Cass means business

Cass Business School
City of London

Faculty of Actuarial Science and Statistics

Automatic, Computer Aided Geometric Design of Free-Knot, Regression Splines

Vladimir K. Kaishev*, Dimitrina S. Dimitrova, Steven Haberman and Richard Verrall

Statistical Research Paper No. 24

August 2004

ISBN 1-901615-81-2

Cass Business School
106 Bunhill Row
London EC1Y 8TZ
T +44 (0)20 7040 8470
www.cass.city.ac.uk

“Any opinions expressed in this paper are my/our own and not necessarily those of my/our employer or anyone else I/we have discussed them with. You must not copy this paper or quote it without my/our permission”.

Automatic, Computer Aided Geometric Design of Free-Knot, Regression Splines

by

Vladimir K. Kaishev*, Dimitrina S. Dimitrova, Steven Haberman and Richard Verrall

Cass Business School, City University, London

Abstract

A new algorithm for Computer Aided Geometric Design of least squares (LS) splines with variable knots, named GeDS, is presented. It is based on interpreting functional spline regression as a parametric B-spline curve, and on using the shape preserving property of its control polygon. The GeDS algorithm includes two major stages. For the first stage, an automatic adaptive, knot location algorithm is developed. By adding knots, one at a time, it sequentially "breaks" a straight line segment into pieces in order to construct a linear LS B-spline fit, which captures the "shape" of the data. A stopping rule is applied which avoids both over and under fitting and selects the number of knots for the second stage of GeDS, in which smoother, higher order (quadratic, cubic, etc.) fits are generated. The knots appropriate for the second stage are determined, according to a new knot location method, called the averaging method. It approximately preserves the linear precision property of B-spline curves and allows the attachment of smooth higher order LS B-spline fits to a control polygon, so that the shape of the linear polygon of stage one is followed. The GeDS method produces simultaneously linear, quadratic, cubic (and possibly higher order) spline fits with one and the same number of B-spline regression functions. The GeDS algorithm is very fast, since no deterministic or stochastic knot insertion/deletion and relocation search strategies are involved, neither in the first nor the second stage. Extensive numerical examples are provided, illustrating the performance of GeDS and the quality of the resulting LS spline fits. The GeDS procedure is compared with other existing variable knot spline methods and smoothing techniques, such as SARS, HAS, MDL, AGS methods and is shown to produce models with fewer parameters but with similar goodness of fit characteristics, and visual quality.

Keywords: spline regression, B-splines, Greville abscissas, CAGD, free-knot splines, control polygon

**Corresponding author's address: Faculty of Actuarial Science and Statistics, Cass Business School, City University, London EC1Y 8TZ, UK. E-mail address: V.Kaishev@city.ac.uk, tel: +44(0)2070408453*

1. Introduction.

Consider a response variable y and an independent variable x, taking values within a certain interval $[a, b]$ and assume there is a functional relationship between x and y of the form

$$y = f(x) + \epsilon, \tag{1}$$

where $f(\cdot)$ is an unknown function and ϵ is a random error variable with zero mean. A problem which arises in a number of statistical applications is to estimate $f(\cdot)$, based on a sample of observations $\{y_i, x_i\}_{i=1}^{N}$.

Different nonparametric smoothing methods for the solution of this problem have been proposed and the related literature is extensive. We will mention here some well known, spatially adaptive smoothing techniques such as: the wavelet shrinkage methods of Donoho and Johnstone (1994, 1995), the variable bandwidth kernel method of Fan and Gijbels (1995), hybrid adaptive splines (HAS) of Luo and Wahba (1997). Another popular approach to smoothing is to use penalized splines, considered by Eubank (1988), Wahba (1990), Marx and Eilers (1996), Rupert and Carroll (2000), Rupert (2002). A third class of methods uses adaptive knot selection procedures, such as stepwise knot inclusion/deletion strategies, to develop variable knot spline regression models. Among the latter are the early work of Smith (1982), the TURBO spline modelling technique of Friedman and Silverman (1989), the MARS method proposed by Friedman (1991), the POLYMARS of Stone et al. (1997), and more recently the minimum description length (MDL) regression splines of Lee (2000) and the spatially adaptive regression splines (SARS) of Zhou and Shen (2001). A different knot removal algorithm for constructing splines with "almost free" knots, chosen from a subset of the data points, was proposed by Lytch and Mørken(1993). Constructing multivariate spline regression and knot location was considered also by Kaishev (1984). A fourth group of works applies reversible jump Markov chain Monte Carlo (RJMCMC) based methods, to develop Bayesian adaptive splines, such as those of Smith and Kohn (1996), Denison at al. (1998) and Biller (2000), in the context of generalized linear models. These procedures simulate tens of thousands of spline models which are then averaged pointwise to produce a resulting estimate of f, but they are associated with a high computational cost and the inconvenience of having the resulting model in a non-explicit form. A stochastic optimization algorithm for "almost free"-knot splines, called adaptive genetic splines (AGS) was recently proposed by Pittman (2002) but the related computational cost is also a concern, as noted by the author.

Constructing free-knot, least squares splines, given a fixed number of knots, has been considered as a nonlinear approximation problem by De Boor and Rice (1968), (see also

De Boor, 2001) and by Jupp (1978). As is well known, the non-linear optimization problem of finding the best least squares free-knot spline approximation may not have a unique solution, and as noted by Jupp (1978), may have potentially a high number of local extrema. The routine of De Boor and Rice (1968), called NEWNOT leads to a possibly locally optimal knot placement, given the number of knots is known. The LS approximation with as few knots as possible has been considered by Hu (1993) and by Schwetlick and Schütze (1995), who combine non-linear optimization with a knot removal and relocation strategy. Reported numerical examples and computer times refer to models with only a small number of knots. However, the computational cost of running such routines may be prohibitive if splines with many more knots are required to fit very unsmooth functions, based on large data sets, such as the HeaviSine, Doppler, Bumps and Blocks examples. The latter were first introduced by Donoho and Johnstone (1994) and are considered here as test examples 4-7 in Section 6. A recent account of free-knot least squares splines and knot selection strategies is provided by Cox et al. (2002).

In conclusion, we note that most of the quoted spline fitting methods of the third and fourth group perform knot placement search, within suitable subsets of candidate knot locations, e.g. the data points $\{x_i\}_{i=1}^N$, and hence are not entirely free-knot splines. They apply either deterministic or stochastic adaptive knot insertion/deletion and relocation strategies, which may suffer from the knot confounding problem, as noted by Zhou and Shen (2001). Moreover, they may become computationally prohibitive for highly unsmooth functions and large data sets (see e.g. Lee 2000). Another drawback of the above mentioned algorithms is that most of them involve parameters whose values need to be subjectively preassigned by the user. For example, in some cases a guess for an initial set of knots is needed or the user is required to set lower and upper bounds for the number of knots to be included in the final fit. However, such choices may significantly affect the performance of the corresponding algorithms and the quality of the resulting fits. A further problem is that some of the methods impose limitations on the data set, e.g., need rescaling so that $x_i \in [0, 1]$, $i = 1, \ldots, N$. The wavelet shrinkage method of Donoho and Johnstone (1994) requires equally spaced $\{x_i\}_{i=1}^N$ with $N = 2^i$, $i = 1, 2, \ldots$. All of the above mentioned procedures do not allow real time, visual control of the entire fitting process, which is a desirable feature in many of the practical applications. As will be seen, the method developed here overcomes the problems that have been identified.

The purpose of this paper is to develop a simple, automatic, numerically efficient, method of recovering the unknown function f, in the form of a free knot, least squares, spline regression model. We take an approach which is new and very different from the existing methods. The basic idea is that fitting a variable knot regression spline to a set of data may be viewed as "geometrically designing" a spline curve, so as to capture the "shape", defined by the data points, in a similar way to designers "drawing" parametric

curves through scattered points on the plane, in Computer Aided Geometric Design (CAGD) applications. As a consequence, our algorithm is very fast, since there is no computationally expensive stochastic or deterministic knot relocation search involved. In order to distinguish the splines produced by this new method from MARS, TURBO splines, POLYMARS, HAS, SARS, AGS, NEWNOT etc, we call our procedure, the method of geometrically designed (GeD) splines, or GeDS in abbreviated form.

We will show that the proposed method of constructing GeD splines may be equally successfully applied to recover both smooth or wiggly functions with highly non-homogeneous smoothness properties over the x range. By recovering f, we mean reproducing it, not only sufficiently accurately (with respect to the related mean squared error), but also with the corresponding estimated curve having desirable visual characteristics. As will be illustrated, our method produces good estimates with an appropriate degree of smoothness, avoiding overfitting or underfitting, for widely varying signal-to-noise ratios. The method is also automatic, since in most of the applications the user needs to input only the data set $\{y_i, x_i\}_{i=1}^{N}$ and run the corresponding code. It is simple, i.e., easy to implement and follow by users with different backgrounds, allowing them to have visual control and understanding of the fitting process and the corresponding output. Finally, we note, that the GeDS method gives rise to a very fast computational algorithm, taking just seconds on a standard PC to recover f, even if it is highly unsmooth and the resulting spline fit involves many knots. We do not aim at necessarily finding spline fits with as few knots as possible, and with optimal knot placement. However, in most of the examples presented in Section 6, the resulting GeD splines have very few knots, producing very low mean squared error (MSE), within the noise level. In the case of the well known Titanium Heat data example, first given in De Boor and Rice (1968) (see our Example 8, Section 6), the MSE of the GeD quadratic spline fit is lower than that for the optimal cubic fit, found by Jupp (1978), both fits having five internal knots.

The paper is organized as follows. In the next section we define the B-spline regression model and show that the latter can be viewed as a parametric B-spline curve. We further recall some important characteristics of B-splines, and B-spline curves, such as their shape preserving and linear precision properties, which will be used in developing GeDS. In Section 3, a new knot location rule, called the averaging method is introduced and shown to have a very good linear precision property. Due to this property this rule is used to define the number and position of the knots of the final GeD spline fits. In Section 4, we introduce stages A and B of the proposed algorithm. Since stage A is essential, its detailed description is given in Section 5. Section 6 contains the results of the numerical performance of GeDS and its comparison with other existing spline fitting procedures. Conclusions and a discussion of the characteristics of GeDS are provided in the closing Section 7.

2. The B-spline regression as a parametric curve

As mentioned earlier, we base our approach to constructing GeD splines on the idea that fitting a variable knot, least squares spline regression to a noisy set of data $\{y_i, x_i\}_{i=1}^{N}$ may be viewed as a process of computer aided geometric design of the shape of a two dimensional parametric curve, guided by the data points whose "true" location on the plane is perturbed by the noise component ϵ. To elaborate further on this idea, recall that a two dimensional parametric curve $\boldsymbol{Q}(t)$ in CAGD is given coordinate-wise as

$$\boldsymbol{Q}(t) = \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix},$$

where t is a parameter , $t \in [a, b]$.

Let us note, that the functional dependence underlying (1) is in fact a functional curve of the form $y = f(x)$ that can be viewed as a special case of a parametric curve for which $x(t) = t$, i.e.,

$$\boldsymbol{Q}(t) = \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix} = \begin{Bmatrix} t \\ f(t) \end{Bmatrix}.$$

In this paper we assume that f is a spline function of degree $n-1$(order n), defined on $[a, b]$, which can be represented as an appropriate linear combination of B-splines of order n. The latter are piecewise polynomial functions of degree $n-1$, defined on the set of knots $\Delta_{k,n} = \{t_i\}_{i=1}^{2n+g_1+\ldots+g_k}$, with

$$\begin{aligned} &t_1 \le t_2 \le \ldots \le t_{n-1} \le t_n < t_{n+1} = \ \ldots = t_{n+g_1} < t_{n+g_1+1} = \ \ldots = t_{n+g_1+g_2} < \\ &\ldots < t_{n+g_1+\ldots+g_{k-1}+1} = \ \ldots = t_{n+g_1+\ldots+g_k} < t_{n+g_1+\ldots+g_k+1} \le \ldots \le t_{2n+g_1+\ldots+g_k}, \end{aligned} \tag{2}$$

where $t_n = a$, $t_{n+g_1+\ldots+g_k+1} = b$, and $1 \le g_i \le n-1$, $i = 1, \ldots, k$ are called the multiplicities of the knots. B-splines coincide with a polynomial of degree $n-1$ at each of the intervals between adjacent, distinct knots and these pieces are smoothly joined at the latter knots, up to their $(n-1-g_i)$-th derivative. In this paper we will use splines with simple knots (of multiplicity one, i.e., $g_i = 1$, $i = 1, \ldots, k$) except for the n left and right most knots which will be assumed coalescent. In this case (2) simplifies to

$$\Delta_{k,n} = \{t_1 = t_2 = \ \ldots = t_n < t_{n+1} < \ldots < t_{n+k} < t_{n+k+1} = \ \ldots = t_{2n+k}\}. \tag{3}$$

Denote by $S_{\Delta_{k,n}}$ the linear space of all n-th order spline functions defined on $\Delta_{k,n}$. In order to express a spline $f \in S_{\Delta_{k,n}}$, one can introduce $p = n + g_1 + \ldots + g_k$ B-splines $N_{i,n}(t)$, $i = 1, \ldots, p$, of order n on $\Delta_{k,n}$, defined through the Mansfield-De Boor-Cox recurrence relation

$$N_{i,0}(t) = \begin{cases} 1 & \text{if} \quad t_i \le t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

$$N_{i,n}(t) = \frac{t-t_i}{t_{i+n-1}-t_i}\, N_{i,n-1}(t) + \frac{t_{i+n}-t}{t_{i+n}-t_{i+1}}\, N_{i+1,n-1}(t). \quad (5)$$

Using B-splines defined on $\Delta_{k,n}$, one can approximate $f(x)$ with a spline function

$$f_{\Delta_{k,n}}(x) = \boldsymbol{\theta}' \boldsymbol{N}_n(x) = \sum_{i=1}^{p} \theta_i\, N_{i,n}(x), \quad (6)$$

where $\boldsymbol{\theta} = (\theta_1, \ldots \theta_p)'$ is a vector of unknown parameters, to be estimated and $\boldsymbol{N}_n(x) = (N_{1,n}(x), \ \ldots, N_{p,n}(x))'$.

If n, k and $\Delta_{k,n}$ are known, one can estimate $\boldsymbol{\theta}$ based on $\{y_i, x_i\}_{i=1}^{N}$ using the least squares method as

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{F}'\boldsymbol{F})^{-1}\boldsymbol{F}'\boldsymbol{Y},$$

where $\boldsymbol{F} = (\boldsymbol{N}_n(x_1), \ \ldots, \boldsymbol{N}_n(x_N))'$, $\boldsymbol{Y} = (y_1, \ \ldots, y_N)'$, and $\boldsymbol{F}'\boldsymbol{F}$ is non-singular, i.e., the Schoenberg-Whitney condition holds. The latter condition states that $\boldsymbol{F}'\boldsymbol{F}$ is non-singular iff each interval $[t_i, t_{i+n}]$ contains at least one observation x_i, i.e., there exist indexes $1 \le l_1 < l_2 < \ldots < l_p \le N$, such that $t_i < x_{l_i} < t_{i+n}$, $i = 1, \ \ldots, p$. Thus, the LS regression spline fit for a fixed $\Delta_{k,n}$ is

$$\hat{f}_{\Delta_{k,n}}(x) = \sum_{i=1}^{p} \hat{\theta}_i\, N_{i,n}(x).$$

However, the degree $n-1$, the number of knots k and their position in $\Delta_{k,n}$ are in general also unknown parameters which need to be determined. As mentioned earlier, such splines are called splines with free or variable knots, for which one of the most important problems is to define the number and location of the knots. In Section 3, we will present an algorithm for the solution of this problem, approaching the regression spline (6) as a parametric curve. Thus, if we view the functional B-spline curve (6) as parametric, we can write

$$\boldsymbol{Q}(t) = \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix} = \begin{Bmatrix} t \\ f_{\Delta_{k,n}}(t) \end{Bmatrix} = \begin{Bmatrix} t \\ \sum_{i=1}^{p} \theta_i\, N_{i,n}(t) \end{Bmatrix}. \quad (7)$$

To develop the GeD spline methodology we will need some of the properties of the B-splines, which have made them the preferred set of basis functions in CAGD, approximation theory and statistics. The first such property of crucial importance for CAGD applications is the partition of unity property.

Property 1 (partition of unity). The sum of all B-splines evaluated at t is equal to one, i.e.,

$\sum_{i=j-n+1}^{j} N_{i,n}(t) = 1$, for any $t \in [t_j, t_{j+1})$, $j = n, \ \ldots, n+k$.

Proof. See, for example De Boor 2001, p. 96. □

Next we give another important property of B-spline curves, called the linear precision property.

Property 2 (linear precision). The following identity holds

$$t = \sum_{i=1}^{p} \xi_i^* \, N_{i,n}(t) \ , \tag{8}$$

where,

$$\xi_i^* = (t_{i+1} + ... + t_{i+n-1}) / (n-1), \, i = 1, \ ..., \ p. \tag{9}$$

Proof. The result follows from Marsden's identity (see e.g. Cohen et al. 2001, Theorems 7.19, 7.14). □

The values ξ_i^* given by (9) are known as the Greville abscissas. In view of the linear precision property (8) we can rewrite (7) as

$$\boldsymbol{Q}(t) = \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix} = \begin{Bmatrix} t \\ f_{\Delta_{k,n}}(t) \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{p} \xi_i^* \, N_{i,n}(t) \\ \sum_{i=1}^{p} \theta_i \, N_{i,n}(t) \end{Bmatrix}. \tag{10}$$

Note that (10) is a subset of the general class of parametric B-spline curves

$$\boldsymbol{Q}(t) = \sum_{i=1}^{p} \boldsymbol{c}_i \, N_{i,n}(t) = \begin{Bmatrix} \sum_{i=1}^{p} \xi_i \, N_{i,n}(t) \\ \sum_{i=1}^{p} \theta_i \, N_{i,n}(t) \end{Bmatrix}, \tag{11}$$

where $\boldsymbol{c}_i = (\xi_i, \theta_i), \ i = 1, \ ..., \ p$, denote the vertexes of the control polygon $\boldsymbol{C}$, of $\boldsymbol{Q}(t)$, called also the control points of $\boldsymbol{Q}(t)$. Note that, due to the partition of unity property of B-splines, any point from a B-spline curve $\boldsymbol{Q}(t)$ in (11) is expressed as a convex, barycentric combination of its control points. This leads us to Property 3.

Property 3 (affine invariance). The parametric B-spline curve $\boldsymbol{Q}(t)$ is affinely invariant.

Proof. The proof follows by the definition of affine invariance (see e.g. Farin 2002). □

We note that $\boldsymbol{Q}(t)$ are also invariant under an affine reparametrization, a property which, as a consequence, holds for GeDS.

Since the set of curves, defined by (10), is a subset of the parametric B-spline curves in (11), each one of them has a control polygon with vertexes (ξ_i^*, θ_i), i.e.,

$$\boldsymbol{Q}(t) = \sum_{i=1}^{p} \boldsymbol{c}_i \, N_{i,n}(t) = \begin{Bmatrix} \sum_{i=1}^{p} \xi_i^* \, N_{i,n}(t) \\ \sum_{i=1}^{p} \theta_i \, N_{i,n}(t) \end{Bmatrix}. \tag{12}$$

A functional B-spline curve $\boldsymbol{Q}(t)$ of order $n = 3$ and its control polygon $\boldsymbol{C}$ are illustrated in Fig. 1. Let us note that the control polygon plays an important role in CAGD since it mimics the shape of its related curve. This is stated by the following property.

Property 4 (shape preserving). The B-spline curve $\boldsymbol{Q}(t)$ has the same shape as its control polygon, i.e, it crosses any straight line no more often than does $\boldsymbol{C}$.

Proof. The result follows by applying the well known Schoenberg's variation diminishing property which states that the number of sign changes in the spline function $\boldsymbol{Q}(t)$ is

not bigger than the number of sign changes in the sequence of its B-spline coefficients θ_i, $i = 1, \ ..., \ p$ (see e.g., De Boor 2001, p. 141). □

In particular, in the linear case ($n = 2$), $\boldsymbol{Q}(t)$ coincides with its control polygon and hence the shape preserving property holds exactly. In the quadratic case ($n = 3$) the curve $\boldsymbol{Q}(t)$, evaluated at the knots $t_3, t_4, \ ..., t_{k+4}$, interpolates $\boldsymbol{C}$ and is tangential to each of its segments, $\boldsymbol{c}_i, \boldsymbol{c}_{i+1}$, dividing it in a proportion $(t_{i+2} - t_{i+1}) : (t_{i+3} - t_{i+2})$, $i = 2, \ ..., k + 2$. This is illustrated by Fig. 1, in the case of $k = 5$, where $\Delta_j = t_{j+1} - t_j$, $j = 3, \ ... k + 3$. In the cubic case ($n = 4$), the spline curve, evaluated at a knot, i.e., $\boldsymbol{Q}(t_{i+3})$ is somewhere within the triangle of points $\boldsymbol{c}_i, \boldsymbol{c}_{i+1} \ \boldsymbol{c}_{i+2}$, $i = 1, 2, \ ..., p$. Hence, the higher the degree, the stronger the curve deviates from its control polygon $\boldsymbol{C}$, but it still remains within the convex hull of $\boldsymbol{C}$, due to the following property.

Property 5 (convex hull). The B-spline curve $\boldsymbol{Q}(t)$ lies within the convex hull of its control polygon, and more precisely, each of its polynomial segments lies within the convex hull of the n control points, defining it.

Proof. The proof follows from the fact that every point of the curve $\boldsymbol{Q}(t)$ of order n is a barycentric combination of n control points, i.e., $\boldsymbol{Q}(t) = \sum_{i=j}^{n+j-1} \boldsymbol{c}_i \, N_{i,n}(t)$, $t \in [t_{n+j-1}, t_{n+j}]$, $j = 1, \ ..., k + 1$. □

The shape preserving and convex hull properties, illustrated in Fig. 1 are an important motivation for developing the GeDS algorithm. The shaded areas in Fig. 1 are examples of convex hulls in the case of a quadratic B-spline curve.

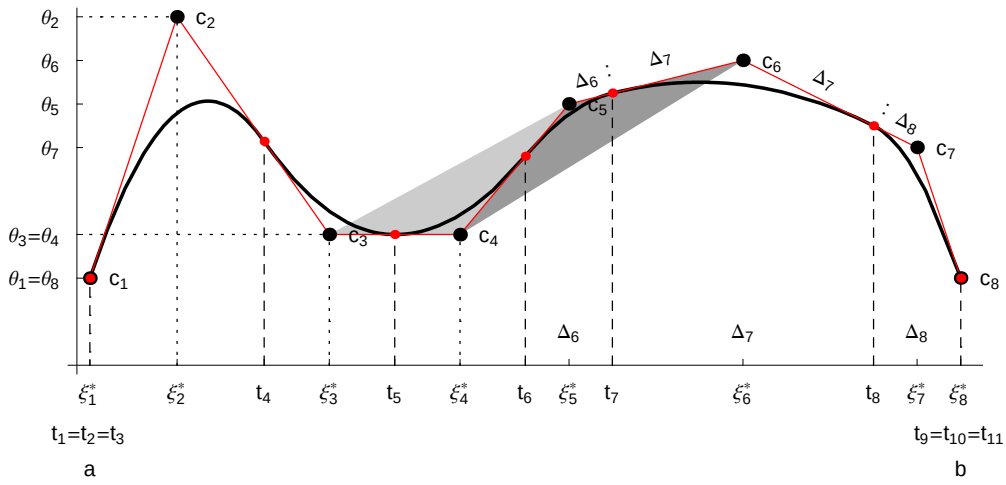

Fig. 1. A quadratic, functional B-spline curve and its control polygon.

As noted above, linear GeDS are optimal, with respect to shape preservation and, as will be seen in Section 6, lead to very accurate fits of f. However they may not be sufficiently smooth, in some applications, where continuous first and higher order derivatives of f are required. Thus, a quadratic B-spline curve is best suited as a compromise between smoothness and shape preservation, having a continuous first derivative and at

the same time being tangential to each of the segments of its control polygon. This makes quadratic splines especially suitable for implementing the GeDS algorithm.

Thus, due to the shape preserving and convex hull properties, the control polygon can be manipulated in order to design the shape of a functional or parametric B-spline curve. We use this approach in solving the problem of recovering the unknown function f from a set of observations $\{y_i, x_i\}_{i=1}^N$ and construct an appropriate control polygon $\boldsymbol{C}$ which captures the shape of the data. Then, having $\boldsymbol{C}$ and the relation between $\boldsymbol{c}_i$ and t_i, given by (9), we define the number and position of the knots of a smooth B-spline curve, which best approximates the data in the LS sense. The procedure of finding the most appropriate control polygon $\boldsymbol{C}$ is the first major stage of our algorithm, explained in details in Section 5. The problem of defining the number and position of the knots of a functional B-spline curve, given its control polygon, comprises the second major stage of the algorithm.

Let us note that, for the functional curves (12), given the knots $\Delta_{k,n}$ and the degree $n-1$, it is always possible to use (9) and find values of the Greville abscissae ξ_i^*. Then, the free parameters to be estimated, based on the data are the y-coordinates of the control points. The latter, called De Boor ordinates, coincide with our unknown spline regression coefficients $\boldsymbol{\theta}$, as seen from (12). Hence, given n and $\Delta_{k,n}$, finding LS estimates of the regression coefficients $\boldsymbol{\theta}$, based on $\{y_i, x_i\}_{i=1}^N$, is equivalent to estimating the location of the y-coordinates of the vertexes of the control polygon in (12). This is an important point, which, along with the other recollected properties of B-spline parametric curves, has allowed us to develop our CAGD approach to constructing free-knot least squares regression splines.

To implement the proposed approach, given a control polygon $\boldsymbol{C}$ and a fixed n, we need to be able to find $\Delta_{k,n}$ of the functional B-spline curve, attached to it. Hence, given the control points $\boldsymbol{c}_i$, if their x-coordinates ξ_i could be obtained as the Greville abscissa values from an appropriate set of knots $\Delta_{k,n}$, one could attach a functional B-spline curve $f_{\Delta_{k,n}}$, on to the polygon $\boldsymbol{C}$, since the linear precision property (8) will be fulfilled. Let us note that conditions (9) are imposed, since we are interested in modeling functional curves in a parametric form. However, for parametric curves, (9) is not required. So, one can arbitrarily choose the set of knots and attach more then one parametric B-spline curve on to a given control polygon.

Unfortunately, it is not always possible to find the knots $\Delta_{k,n}$, given the x-coordinates, ξ_i, of the control points, so that (9) is satisfied. It can be seen that expressions (9) form an over-determined system of equations, with constraints on the knots, given by the definition of $\Delta_{k,n}$. Since $\xi_1 = a$ and $\xi_p = b$, the system (9) contains $k+n-2$ equations and k ordered, unknown knots. In the next section we propose a method, which overcomes this difficulty and expresses the set of internal knots, through ξ_i, $i = 1, \ldots, p$, so that the linear precision property (8) of B-spline curves holds, at least approximately.

3. Positioning of the knots

In this section, we present a method, called averaging knot location method. It allows to avoid the problem of solving system (9) with respect to t_{i+n}, $i = 1, \ ..., \ k$ and at the same time provides a set of knots $\Delta_{k,n}$, such that the B-spline curve $f_{\Delta_{k,n}}$ approximately obeys the linear precision property (8). This implies that, for given ξ_i of $\boldsymbol{C}$, the averaging knot location method produces $\Delta_{k,n}$, such that the Greville abscissas ξ_i^*, obtained from $\Delta_{k,n}$, are very close to ξ_i of $\boldsymbol{C}$.

1) The averaging knot location method

Choose the internal knots in $\Delta_{k,n}$ as the averages of the x-coordinates of the vertexes of the control polygon $\boldsymbol{C}$, i.e.,

$$t_{i+n} = (\xi_{i+1} + ... + \xi_{i+n-1}) / (n-1), \qquad i = 1, \ ..., \ k. \tag{13}$$

The following proposition establishes an important property of rule (13), which will be used throughout the sequel.

Proposition 1. The averaging knot location method (13) is affinely invariant.

Proof. Note that, according to (13), an internal knot t_{i+n} is a convex, barycentric combination of $\xi_{i+1}, \ ..., \ \xi_{i+n-1}$. Hence, the assertion of the proposition follows, since affine transformations leave barycentric combinations invariant. □

Now, we investigate the extent to which the averaging knot location method preserves the linear precision property (8).

Proposition 2. The deviation $\delta(t) := \ | \sum_{i=1}^{p} \xi_i^* \, N_{i,n}(t) - \sum_{i=1}^{p} \xi_i \, N_{i,n}(t) \ |$ of the spline function $\sum_{i=1}^{p} \xi_i \, N_{i,n}(t)$, with knots given by (13), from the straight line $t \equiv \sum_{i=1}^{p} \xi_i^* \, N_{i,n}(t)$, $t \in \ [a, \, b]$ is bounded by

$$\delta(t) \le \max_{j \in \{2,...,p-1\}} | \, \xi_j^* - \xi_j \, |. \tag{14}$$

Proof. Note that

$$\delta(t) = \ | \sum_{i=1}^{p} (\xi_i^* - \xi_i) \, N_{i,n}(t) \ | \ \le \sum_{i=1}^{p} | \, (\xi_i^* - \xi_i) \, N_{i,n}(t) \, |$$

$$\le \max_{j \in \{2,...,p-1\}} | \, \xi_j^* - \xi_j \, | \sum_{i=1}^{p} N_{i,n}(t) \le \max_{j \in \{2,...,p-1\}} | \, \xi_j^* - \xi_j \, |$$

where we have applied the partition of unity property of $N_{i,n}(t)$, $i = 1, \ ..., \ p$. □

In order to assess the accuracy of the bound (14) and illustrate the extent to which the averaging knot location method preserves the linear precision property of B-spline curves, we have randomly generated abscissa values ξ_j for three fixed numbers of vertices p, equal respectively to 6 ($k = 3$), 11 ($k = 8$) and 23 ($k = 20$), in the quadratic case ($n = 3$). The number of simulations for each value of p is 1000. The corresponding

thousand graphs of $\sum_{i=1}^{p} \xi_i N_{i,n}(t)$, $t \in [0, 1]$ with knots defined by (13), have been plotted in Fig. 2 (a), (b) and (c).

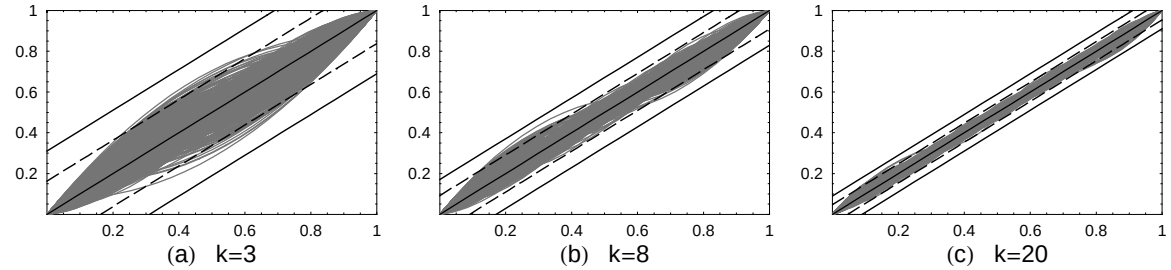

Fig. 2. Graphs of 1000 simulations of $\sum_{i=1}^{p} \xi_i N_{i,n}(t)$, with $\Delta_{k,3}$ according to (13) and estimates of $\hat{\delta}_{0.95}^{\max}$ and $\hat{b}_{0.95}^{\max}$ for (a) $p = 6$ $(k = 3)$, $\hat{\delta}_{0.95}^{\max} = 0.16$, $\hat{b}_{0.95}^{\max} = 0.31$; (b) $p = 11$ $(k = 8)$, $\hat{\delta}_{0.95}^{\max} = 0.09$, $\hat{b}_{0.95}^{\max} = 0.17$; (c) $p = 23$ $(k = 20)$, $\hat{\delta}_{0.95}^{\max} = 0.05$, $\hat{b}_{0.95}^{\max} = 0.09$.

In Fig. 2, two corridors are also shown. The first, defined by the dashed lines, is based on the 95 sample percentile of the $\max_t \delta(t)$, denoted by $\hat{\delta}_{0.95}^{\max}$. The second corridor (the solid lines) is based on the bound (14), denoted by $\hat{b}_{0.95}^{\max}$. As can be seen from Fig.2, the maximum deviation of $\sum_{i=1}^{p} \xi_i N_{i,n}(t)$ from the straight line t is reasonable, and decreases as the number of knots increases. Thus, the higher the number of knots, the better rule (13) allows for the linear precision property of a B-spline curve to be preserved. Similar conclusions are found to hold for the cubic case ($n = 4$), applying both $\hat{\delta}_{0.95}^{\max}$ and $\hat{b}_{0.95}^{\max}$.

We have explored also other possible methods for defining the knots through the coordinates of the control points $\boldsymbol{c}_i$, $i = 1, \ldots, p$. In order to formulate these methods, we have applied ideas which are similar to those used in CAGD to define rules for choosing parameter values, that correspond to some points on the plane, to be interpolated by a parametric B-spline curve. In this case, several alternative methods, such as the uniform, the chord length, and the centripetal methods have been proposed in the CAGD literature (see Farin 2001). According to these methods, the parameter values are chosen to be proportional to the distances between the data points which are interpolated. However, our purpose here is different, in that we seek to express the knots of a functional B-spline curve which approximates a set of data through the control points. So, we define the following alternatives to the latter methods.

2) The uniform method

$$t_{i+n} = a + i \, \tfrac{b-a}{k+1}, \; i = 1, \ldots, k \tag{15}$$

3) The "Chord Length" method

$$t_{i+n} = a + (b-a)\left(\tfrac{L_{i+1}+\ldots+L_{i+n-1}}{n-1}\right)/L, \quad i = 1, \ldots, k, \tag{16}$$

where $L = \sum_{j=2}^{p} \|c_j - c_{j-1}\| = \sum_{j=2}^{p} \sqrt{(\xi_j - \xi_{j-1})^2 + (\theta_j - \theta_{j-1})^2}$

and $L_l = \sum_{j=2}^{l} \|c_j - c_{j-1}\|$, $l = 2,\ ...,\ p-1$.

4) The "Centripetal" method

$$t_{i+n} = a + (b-a)\,(\tfrac{L_{i+1}+...+L_{i+n-1}}{n-1})\,/\,L, \quad i = 1,\ ...,\ k, \tag{17}$$

where $L = \sum_{j=2}^{p} (\|c_j - c_{j-1}\|)^{0.5}$ and $L_l = \sum_{j=2}^{l} (\|c_j - c_{j-1}\|)^{0.5}$, $l = 2,\ ...,\ p-1$.

We have investigated these three rules, as alternatives to the averaging knot location method. Their ability to preserve the linear precision property (8) is illustrated in Fig 3. A comparison of Fig. 3 with Fig. 2 (c), shows that these methods are considerably worse than the averaging knot location method. Note, that rules 2) and 3) are not affine invariant and use both the x and y coordinates of $\boldsymbol{c}_i$, in contrast to the averaging knot location method, which is affine invariant and uses only the x-coordinates ξ_i, $i = 1,\ ...,\ p$ of the vertexes of $\boldsymbol{C}$.

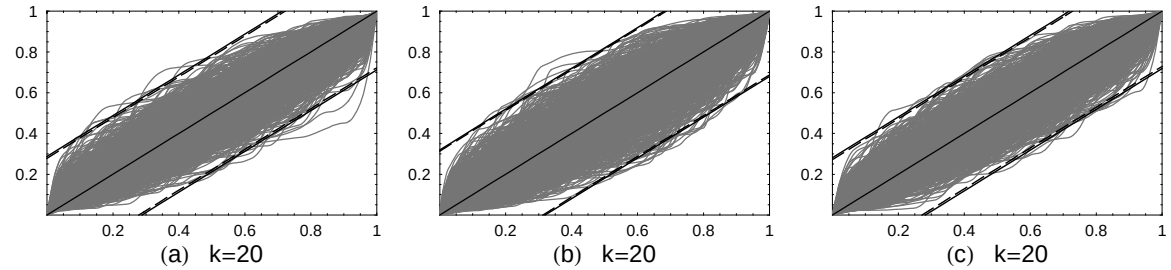

Fig. 3. Graphs of 1000 simulations of $\sum_{i=1}^{p} \xi_i N_{i,n}(t)$ for $p = 23$ $(k = 20)$ with $\Delta_{k,3}$ defined according to: (a) the uniform method (15), $\hat{\delta}_{0.95}^{\max} = 0.28$, $\hat{b}_{0.95}^{\max} = 0.29$; (b) the chord length method (16), $\hat{\delta}_{0.95}^{\max} = 0.31$, $\hat{b}_{0.95}^{\max} = 0.32$; (c) the centripetal method (17), $\hat{\delta}_{0.95}^{\max} = 0.27$, $\hat{b}_{0.95}^{\max} = 0.28$.

4. The GeDS algorithm

In order to solve the problem of recovering the unknown function in (1) and construct a GeD spline, as a free-knot, least squares B-spline regression model of order $n \geq 2$, we apply the CAGD ideas and properties of parametric B-spline curves, as described in Section 2. Thus, in CAGD, in order to design a curve, the number and position of its control points are interactively manipulated, so as to reach a desirable position of the curve in the plane. In the latter design approach, it is essential to manipulate the control polygon in such a way that it represents a rough version (i.e. a linear approximation) of the smooth shape which has to be drawn. A smooth B-spline parametric curve is "attached" to the control polygon on each step of the interactive design process until the desirable shape and position of the curve is achieved. The use of the control polygon in the design is based on the shape preserving property of parametric spline curves, discussed in Section 2.

We transfer these CAGD ideas to the context of non-parametric regression smoothing, in order to develop our new method for the construction of "geometrically designed", free knot, least squares B-spline regression curves, i.e., GeDS. The method includes two major stages, A and B. In stage A, a LS linear spline fit is constructed, in order to obtain the "geometric form" of the data. For this purpose, a new, spatially adaptive procedure for automatic knot insertion, equipped with an appropriate stopping rule, is introduced. The LS linear fit is then used in stage B as a guideline for designing the shape of a smoother, quadratic, cubic (or higher order) LS spline model. We note that the knots are determined as the averages of the knots of the linear spline fit, applying (13). This distinguishes our GeDS algorithm from other existing spline methods and makes it very fast, since no time consuming, knot insertion-deletion schemes, or other simulation and search algorithms are involved. As will be seen from the examples, stages A and B are sufficient to obtain an accurate fit to the data.

In what follows, we give further details of the two stages of the proposed procedure for constructing GeDS.

Stage A. ***Construction of a free knot, LS linear B-spline fit by an automatic knot insertion algorithm.***

In this first stage, an automatic knot insertion algorithm is applied to construct a free-knot, least squares, linear (order $n = 2$) B-spline curve (polygon), which reproduces the "shape" of the data set $\{y_i, x_i\}_{i=1}^{N}$. The algorithm may be given the following geometric interpretation. It starts from an LS fit, in the form of a straight line segment. The latter is then sequentially "broken" into a piecewise linear LS fit, by adding knots, one at a time, at some points, where the fit deviates most from the "shape" of the data, according to a measure based on appropriately defined clusters of residuals. A stopping rule is introduced, which allows us to determine the appropriate number and location of the knots of the linear spline fit and thus, avoid over- or under-fitting. Note that the LS linear B-spline fit $\hat{f}_{\Delta_{k,2}}(x)$, produced in this way, coincides with its control polygon, hence its knots $\Delta_{k,2}$ coincide with the abscissas of its control points, i.e., $t_{i+1} = \xi_i$, $i = 1, ..., p$, $p = k + 2$. So, as it will be illustrated in the next section, the linear GeD spline fit $\hat{f}_{\Delta_{k,2}}(x)$ is a sufficiently accurate reconstruction of the unknown function, given that no further smoothness is required. If a smoother fit is required, a higher order GeD spline is constructed in stage B of the GeDS procedure. A formal description of the algorithm for stage A is given in Section 5.

Stage B. ***Designing the shape of a higher order (quadratic, cubic etc.) LS spline curve, via the LS B-spline polygon of stage A.***

For $n = 3, 4, ...$ we apply the averaging knot location method (13), and choose the k internal knots t_{i+n}, $i = 1, ..., k$ in $\Delta_{k,n}$, as the averages of abscissa values ξ_{i+1}, $i = 1, ..., k$ of the vertexes of the LS B-spline polygon, $\hat{f}_{\Delta_{k,2}}(x)$, produced in stage A.

Based on $\Delta_{k,n}$ we then construct a higher order (quadratic, cubic etc.) LS B-spline regression curve, $\hat{f}_{\Delta_{k,n}}(x)$, fitting the data. The latter fit has a control polygon with vertexes, whose y-coordinates are the LS regression estimates $\hat{\theta}_i$, $i = 1, \ldots, p$ and whose x-coordinates are the Greville abscissas ξ_i^*, obtained from the knots $\Delta_{k,n}$, applying (9). We note that the p vertexes of the two polygons, the control polygon of the LS higher order B-spline fit $\hat{f}_{\Delta_{k,n}}(x)$, and the LS linear B-spline fit, $\hat{f}_{\Delta_{k,2}}(x)$, will be correspondingly close to each other. Their x-coordinates ξ_i and ξ_i^* are close, due to the linear precision property of the averaging knot location method (see Fig 2). Their y-coordinates, $\hat{f}_{\Delta_{k,2}}(\xi_i)$ and $\hat{\theta}_i$, are close, since $\hat{f}_{\Delta_{k,2}}(\xi_i)$ and $\hat{f}_{\Delta_{k,n}}(\xi_i^*)$ are close as LS fits to $\{y_i, x_i\}_{i=1}^N$, evaluated at the close x locations ξ_i and ξ_i^*, and $\hat{f}_{\Delta_{k,n}}(\xi_i^*) \approx \hat{\theta}_i$. For a proof of the fact that $\hat{f}_{\Delta_{k,n}}(\xi_i^*) \approx \hat{\theta}_i$ see e.g., Cohen et al. (2001), p. 281.

In this way, we assure that the control polygon of $\hat{f}_{\Delta_{k,n}}(x)$ is close to the LS B-spline polygon $\hat{f}_{\Delta_{k,2}}(x)$. But, due to its shape preserving property (see Section 2), the fit $\hat{f}_{\Delta_{k,n}}(x)$ will have the same shape as its control polygon, hence will be close to the shape of the LS B-spline polygon $\hat{f}_{\Delta_{k,2}}(x)$, which follows the shape of the data. But since, applying the averaging knot location method (13), more knots are inserted at locations where $\hat{f}_{\Delta_{k,2}}(x)$ is more wiggly and less knots at its smoother segments, we guarantee that more knots are placed where the data exhibits more variation. In this way we assure that $\hat{f}_{\Delta_{k,n}}(x)$ has appropriately located set of knots and adequately approximates the data. This is the basic CAGD idea underlying stage B of the proposed GeDS method. It allows us to avoid complex and time consuming knot optimization procedures. As will be illustrated by the examples in Section 6, the resulting fits have good visual quality and appropriate goodness of fit measure.

Remark: Stages A and B are sufficient to produce a very good quality spline fit with a reasonably small number of knots as seen from the examples, given in Section 6. However, since GeDS does not produce optimally placed knots, in some applications, a Fibonacci optimization search applied sequentially to each knot, produced after Stage B, may give some improvement of the MSE (see Example 8, Section 6).

5. An automatic, knot insertion algorithm for free-knot, LS linear B-spline regression

In view of the importance of stage A of GeDS, this section contains a detailed description.

Step 1. Set $n = 2$ and $k = 0$, i.e, the starting set of knots is $\Delta_{0,2} = \{t_i\}_{i=1}^4$ with $t_1 = t_2 = a < b = t_3 = t_4$ and find the LS B-spline fit, in the form of the straight line

$$\hat{f}_{\Delta_{0,2}}(x) = \hat{\theta}_1 N_{1,2}(x) + \hat{\theta}_2 N_{2,2}(x).$$

Find the residuals $r_i \equiv r(x_i) = y_i - \hat{f}_{\Delta_{0,2}}(x_i)$, $i = 1, \ldots, N$ and calculate the residual sum of squares $\mathrm{RSS}(k) = \sum_{i=1}^{N} r_i^2$ and the mean squared error $\mathrm{MSE}(k) = \mathrm{RSS}(k)/N$ of the fit with k internal knots. Since the i-th residual $r(x_i)$, is a function of x_i, $i = 1, 2, \ldots, N$ we will refer to x_i as the x-value of the i-th residual.

Step 2. Group the consecutive residuals r_i, $i = 1, \ldots, N$ into clusters by their sign, i.e., find a number l, $1 \le l \le N$ and a set of integer values $d_j > 0$, $j = 1, \ldots, l$ such that

$$\begin{aligned}&\mathrm{sign}(r_1) = \ldots = \mathrm{sign}(r_{d_1}) \neq \mathrm{sign}(r_{d_1+1}) = \mathrm{sign}(r_{d_1+2}) = \ldots = \mathrm{sign}(r_{d_1+d_2}) \neq \\ &\ldots \neq \mathrm{sign}(r_{d_1+d_2+\ldots+d_{l-1}+1}) = \mathrm{sign}(r_{d_1+d_2+\ldots+d_{l-1}+2}) = \ldots = \mathrm{sign}(r_{d_1+d_2+\ldots+d_l}),\end{aligned}$$

and $\sum_{j=1}^{l} d_j = N$. Note that the clusters are formed and numbered consecutively, following the order of the residuals, i.e., the order of their x-values $x_1 < x_2 < \ldots < x_N$.

Step 3. For each of the l clusters of residuals of identical signs, calculate the "within-cluster" mean residual value

$$m_j = \left(\sum_{i=1}^{d_j} r_{d(j)+i} \right) \Big/ d_j, \qquad j = 1, \ldots, l,$$

where $d(j) = d_1 + d_2 + \ldots + d_{j-1}$ and the "within-cluster" range ξ_j, defined as the difference between the right-most and the left-most x-value of the residuals, belonging to the j-th cluster, i.e., $\xi_j = x_{d(j+1)} - x_{d(j)+1}$, $j = 1, \ldots, l$. Throughout the sequel, we will need the more general notation for $d(j) = \sum_{i<j} d_i$ to denote partial sums of non-ordered values d_i, for which $i < j$ and we will call the interval between the right-most and the left-most x-value of the residuals, belonging to the j-th cluster, i.e., $[x_{d(j)+1}, x_{d(j+1)}]$, the "within-cluster" interval.

Step 4. Find

$$m_{\max} = \max_{1 \le j \le l} (m_j) \tag{18}$$

$$\xi_{\max} = \max_{1 \le j \le l} (\xi_j) \tag{19}$$

and calculate, correspondingly, the normalized "within-cluster" mean and range values $m'_j = m_j / m_{\max}$ and $\xi'_j = \xi_j / \xi_{\max}$, so that $0 < m'_j \le 1$, $0 < \xi'_j \le 1$. Note that the equalities (18) and (19) will not necessarily be fulfilled for one and the same cluster index j, i.e., the two maximums $m_{\max}$ and $\xi_{\max}$ may in general be attained for 2 different clusters with indexes $j_m \neq j_\xi$.

Step 5. Calculate the cluster weights

$$w_j = \beta m'_j + (1 - \beta) \xi'_j, \quad j = 1, \ldots, l, \tag{20}$$

where, β is a real valued parameter, $0 \le \beta \le 1$. The value w_j can serve as a measure, attached to the j-th cluster of residuals of identical sign, which measures the deviation of the least squares linear B-spline regression polygon $\hat{f}_{\Delta_{k,2}}(x)$ from the j-th cluster.

Obviously, the weight w_j itself is a weighted sum of the normalized "within-cluster" mean and "within-cluster" range values and the weight β is one of the parameters, whose value will need to be chosen by the user at the start of Stage A.

Step 6. Order the clusters in descending order of their weights w_j, $j = 1, \ldots, l$, i.e., create a list of corresponding cluster indexes $\{j_1, j_2, \ldots, j_l\}$ such that $w_{j_1} \geq w_{j_2} \geq \ldots \geq w_{j_l}$. In the case where some clusters have coincident weights, they are ordered in descending order of their "within-cluster" means. If the latter coincide, the order between the clusters is set, according to the descending values of the "within-cluster" ranges. In the case of coincident "within-cluster" ranges, the clusters are ordered with respect to the number of residuals (of identical sign) in each cluster. Finally, if all of the listed characteristics of some of the clusters are identical, they are ordered in decreasing order with respect to the x-value of the right most residual in the cluster. These proposed ways of ordering the clusters are reasonable and meaningful, since they all characterize, in one way or another, how much the current least squares linear spline fit $\hat{f}_{\Delta_{k,2}}(x)$ deviates from each of the clusters. Thus, to improve $\hat{f}_{\Delta_{k,2}}(x)$, we insert a new knot, at an appropriate location, in the "within-cluster" interval of x-values, corresponding to the j_1-th cluster. Since, in general, even equality of the w_j is relatively unlikely, the ordering of the clusters is based practically on the ordering of their weights w_j. The precise definition of the new knot placement criterion is given in the next step.

Step 7. Check whether there is already a knot in the "within-cluster" interval of the j_1-th cluster with highest rank, according to the ordering in Step 6, i.e., check whether

$$t_i \in [x_{d(j_1)+1}, x_{d(j_1)+d_{j_1}}] , \tag{21}$$

for each internal knot $t_i \in \Delta_{k,2}$, $i = 3, \ldots, k+2$.

If there is already a knot in the "within-cluster" interval of the j_1-th cluster, the check is repeated for the cluster with index j_2, and so on until the first cluster, with index j_s, say, in the ordering of clusters is found, whose "within-cluster" interval does not contain a knot then insert a new knot t^* at

$$t^* = \left(\sum_{i=d(j_s)+1}^{d(j_s)+d_{j_s}} r_i \, x_i \right) \Big/ \left(\sum_{i=d(j_s)+1}^{d(j_s)+d_{j_s}} r_i \right), \tag{22}$$

Note that (22) is a convex combination of the x-values of the residuals in the cluster with index j_s, whose "within cluster" interval does not contain a knot. The new knot position can be viewed as the weighted average of the x-values of the residuals in the j_s-th cluster, the weights being the normalized values of the residuals. Thus, we use the information about the x-values for which the fit is worse, in that it departs most strongly from the data. Note that, in defining the position of the new knot, we use information about the values of both the independent variable x and the response variable y.

After the location of the new knot t^* is found, the Schoenberg-Whitney condition is checked with respect to $\Delta_{k,2} \bigcup \{t^*\}$. If this condition is violated, the new knot is placed at the first cluster for which it holds and (21) does not hold. If there are no such clusters the algorithm exits from Stage A with $\Delta_{k,2}$. Otherwise, the set of knots $\Delta_{k,2}$ is updated, by adding t^* to it, i.e., $\Delta^*_{k+1,2} := \Delta_{k,2} \bigcup \{t^*\}$, the number of interior knots k is increased by one and Step 8 is executed.

Step 8. Find the least squares linear B-spline fit

$$\hat{f}_{\Delta^*_{k+1,2}}(x) = \sum_{i=1}^{p} \hat{\theta}_i N_{i,2}(x).$$

Since $\Delta^*_{k+1,2}$ contains the new knot, the number of B-splines p will increase by one.

Step 9. Calculate the MSE$(k+1)$ for $\hat{f}_{\Delta^*_{k+1,2}}(x)$. Note that $\Delta_{k,2} \subset \Delta^*_{k+1,2}$ implies that $S_{\Delta_{k,2}} \subset S_{\Delta^*_{k+1,2}}$ hence $\hat{f}_{\Delta_{k,2}}(x) \in S_{\Delta^*_{k+1,2}}$ and applying the orthogonality property of least squares estimation it is easy to show that

$$\sum_{i=1}^{N} \left(y_i - \hat{f}_{\Delta_{k,2}}(x_i)\right)^2 = \sum_{i=1}^{N} \left(y_i - \hat{f}_{\Delta^*_{k+1,2}}(x_i)\right)^2 + \sum_{i=1}^{N} \left(\hat{f}_{\Delta^*_{k+1,2}}(x_i) - \hat{f}_{\Delta_{k,2}}(x_i)\right)^2. \qquad (23)$$

Equality (23) implies that MSE$(k+1) <$ MSE(k). It is obvious also that MSE(k) will converge to zero as $k+n \longrightarrow N$ since, when $k+n=N$ the fit interpolates the data. The insertion of the new knot t^* at a location, where the fit deviates most from the data, assures that the decrement of the MSE, will be significantly big, although not necessarily maximal. Equalities (22) and (23) give rise to the rule for exit from Stage A of the algorithm, given next.

Step 10. If the set of knots $\Delta^*_{k+1,2}$ contains less than q internal knots, for some given value of q, then the algorithm goes back to Step 2. If this is not the case and $\Delta^*_{k+1,2}$ contains q or more internal knots then the ratio

$$\alpha = \text{MSE}(k+1) / \text{MSE}(k+1-q) \qquad (24)$$

is calculated and if $\alpha > \alpha_{\text{exit}}$, an exit from Stage A of the algorithm is performed. The value α_{exit} is chosen ex ante to be close to 1, since the ratio α will be close to zero if the fit has improved significantly and will tend to 1 if no improvement has been achieved on the last $q+1$ consecutive iterations and the corresponding values of the MSE have stabilized. Our experience has shown that the rule (24) works well as a model selector with $q=2$, i.e., stabilization with respect to MSE$(k-1)$, MSE$(k+1)$ is sufficient to exit from Stage A with the appropriate number of knots. Hence, q has been fixed equal to two.

This completes the description of Stage A of GeDS. To summarize, there are only two parameters β and α_{exit} associated with the GeDS algorithm. Their choice is discussed in Section 6.1.

6. GeDS in action

The implementation of the GeDS algorithm has been carried out using *Mathematica*. We have run the GeDS *Mathematica* code for all test examples on a standard PC (Pentium IV, 1.4 Ghz, 512 RAM). The code is available, upon request to the corresponding author.

■ 6.1. Input to the GeDS *Mathematica* program

In order to run the program, it is necessary to input only the set of data $\{x_i, y_i\}_{i=1}^{N}$. The two parameters, $\alpha_{\text{exit}} \in [0, 1]$ and $\beta \in [0, 1]$, defined correspondingly in steps 10 and 5 of Section 5, by means of which the exit from GeDS can be controlled, have preassigned values, which in general need not be re-set. The parameter α_{exit} is related to the stopping rule, which determines when to exit from Stage A, i.e., the number and location of the knots of the final LS linear B-spline fit. The parameter β is related to the residual measure (20) and its choice depends on the wiggliness of the recovered function and the level of the noise ϵ. In the Normal case, $\epsilon \sim N(0, \sigma_\epsilon)$, the noise level is defined by the variance σ_ϵ^2. As will be illustrated, most of the examples are run with the two parameters having the preassigned values $\alpha_{\text{exit}} = 0.9$, $\beta = 0.5$ and this produces very good results. Choices of $\alpha_{\text{exit}} \in [0, 0.7]$ make the algorithm exit after the first few steps which, for most functions, does not lead to an adequate resulting fit.

The choice of β depends on the level of the signal-to-noise ratio (SNR), $\text{SNR} = (\text{var}(f))^{0.5} / \sigma_\epsilon$ and on the degree of smoothness of f. As will be seen, in most of the numerical examples, the appropriate value of β was 0.5, which means that the "within-cluster" mean and range can be considered equally important components of the weights w_j, $j = 1, ..., l$. However, based on our experience, when the SNR is high and f is smooth recommended values are $\beta \in [0.5, 0.6]$, $\alpha_{\text{exit}} = 0.9$. If the SNR is high and f is a wiggly function then the recommended choice is $\beta \in [0.5, 0.6]$, $\alpha_{\text{exit}} \in [0.99, 0.999]$, since otherwise underfit may result. In the case when SNR is low and f is smooth, one may use $\beta \in [0.4, 0.5]$, $\alpha_{\text{exit}} \in [0.9, 0.99]$. It is known that, when the SNR is low and the underlying function is very unsmooth, recovering f is very difficult and different choices of β and α_{exit} may need to be attempted.

■ 6.2. Numerical results

In order to facilitate comparison of GeDS with existing smoothing methods, we have simulated data using the functions given in Table 1, which have been widely used in testing other existing smoothing procedures.

Table 1. Summary of test functions.

Function	Specification
1	$f_1(x) = \frac{10x}{1+100x^2}$
2	$f_2(x) = (4x-2) + 2e^{-16(4x-2)^2}$
3	$f_3(x) = \sin(8x-4) + 2e^{-16(4x-2)^2}$
HeaviSine	$f_4(x) = 4\sin(4\pi x) - \mathrm{sgn}(x-0.3) - \mathrm{sgn}(0.72-x)$
Doppler	$f_5(x) = \sqrt{x(1-x)}\,\sin\left(\frac{2\pi(1+\epsilon)}{(x+\epsilon)}\right),\ \epsilon = 0.05$
Bumps	$f_6(x) = \sum_j h_j \left(1 + \left\lvert\frac{x-s_j}{w_j}\right\rvert\right)^{-4}$, $\{h_j\} = \{4, 5, 3, 4, 5, 4.2, 2.1, 4.3, 3.1, 5.1, 4.2\}$ $\{s_j\} = \{0.1, 0.13, 0.15, 0.23, 0.25, 0.40, 0.44, 0.65, 0.76, 0.78, 0.81\}$ $\{w_j\} = \{0.005, 0.005, 0.006, 0.01, 0.01, 0.03, 0.01, 0.01, 0.005, 0.008, 0.005\}$
Blocks	$f_7(x) = \sum_j h_j \frac{1+\mathrm{sgn}(x-s_j)}{2}$, $\{h_j\} = \{4, -5, 3, -4, 5, -4.2, 2.1, 4.3, -3.1, 2.1, -4.2\}$ $\{s_j\} = \{0.1, 0.13, 0.15, 0.23, 0.25, 0.40, 0.44, 0.65, 0.76, 0.78, 0.81\}$

The data sets, used to test GeDS were simulated by adding noise to each of the seven functions, as given in Table 1. As seen, we have included examples testing GeDS for different values of SNR, and for various characteristics of the data set: small and large sample sizes, x-values in a grid or uniformly generated within different intervals $x \in [a, b]$. In all examples, the noise has a Normal distribution, with the exception of Example 1, where the noise is uniformly distributed. Note also that the test functions included in Table 2 possess different smoothness properties: some of them are relatively smooth, others very wiggly.

Table 2. Summary of examples used to test GeDS.

Example No	***Function (data)***	***Interval***	***Sample size, N***	***Data*** $x_i,\ i = 1, \ldots, N$	***Noice level,*** σ_ϵ	***SNR***
1	$f_1(x)$	$[-2, 2]$	90	$x_i = -2 + \frac{(2-(-2))}{89}(i-1)$	$U(-0.05, 0.05)$	–
2	$f_2(x)$	[0, 1]	256 150	$U(0, 1)$	0.6, 0.4, 0.25 0.25	2, 3, 5 5
3	$f_3(x)$	[0, 1]	256	$U(0, 1)$	0.3	3
4	HeaviSine	[0, 1]	2048	$x_i = \frac{1}{2047}(i-1)$	1	7
5	Doppler	[0, 1]	2048	$x_i = \frac{1}{2047}(i-1)$	1	7
6	Bumps	[0, 1]	2048	$x_i = \frac{1}{2047}(i-1)$	1	7
7	Blocks	[0, 1]	2048	$x_i = \frac{1}{2047}(i-1)$	1	7
8	Titanium Heat Data	[595, 1075]	49	$x_i = 595 + 10(i-1)$	–	–

In order to compare the quality of the fits produced by GeDS to those given by other authors, we use the MSE, defined with respect to the true function f, rather than to the data:

$$\text{MSE} = \left(\sum_{i=1}^{N} \left(f(x_i) - \hat{f}_{\Delta_{k,n}}(x_i) \right)^2 \right) \Big/ N.$$

Note that, in practice, the underlying function is unknown and a set of observations is fitted. For this reason, we give also the L_2-error of approximation, defined as $\sqrt{\text{RSS}}$. However, for fair comparison between the smoothing methods, one would need all model parameter values, such as, number of knots (regression functions) and degree of the spline fits etc., which often are not reported in full. The Titanium Heat Data example is appropriate to compare different smoothing methods since the data are real and DeBoor and others have published the number and position of the knots and the degree of their spline fits. For fair comparison of the speed of computation one would need to implement all available methods using the same hardware and software, and test them on entirely identical simulated data sets. Such a comparison is outside the scope of this paper.

We have run GeDS with 400 simulated data sets for Examples 1-3 and 31 data sets for Examples 4-7. This allows us to compute the median of the MSE, obtained using GeDS, and compare it with the MSE medians given by other authors. However, in order to illustrate how GeDS performs, in each example we have used a single data set randomly chosen among the simulated data sets.

We compare most of our results, except those for the Blocks and Bumps examples, with the results of Luo and Wahba (1997) since, along with the median MSE values for their fits, they give also the order and the number of the basis functions. We have excluded the Bumps and Blocks since Luo and Wahba (1997) use versions of these functions which differ from ours, i.e., from those proposed by Donoho and Johnstone (1994).

Example 1. We start by testing GeDS on recovering the function f_1, which appears in Schwetlick and Schütze (1995). Our 400 simulated data sets have the same characteristics as the data set of Schwetlick and Schütze (1995) (see Table 2), so we are able to compare our results with theirs. Graphs of the linear B-spline fits, produced on each consecutive iteration in Stage A of GeDS, preceding the final one, are given in Fig. 4. As can be seen, the initial, straight line fit, given in Fig. 4 (a), is sequentially improved by adding knots, one at each step, to reach the fit given in Fig. 5 (a), which can not be significantly improved by adding more knots. Applying the averaging knot selection method (13) to the knots of the final linear fit of Stage A, the set of knots of the quadratic and cubic LS spline fits are defined. These fits, resulting from stage B of GeDS, are correspondingly plotted in Fig. 5 (b) and (c). The closeness of the control polygons of the final, linear, quadratic and cubic fits, is illustrated in Fig. 5 (d). It can be seen from Fig. 5 (a), (e) and (f), that the shape preserving property of B-splines holds and the fits follow the shape of their control polygons but they move away from them as the order of the B-splines increases.

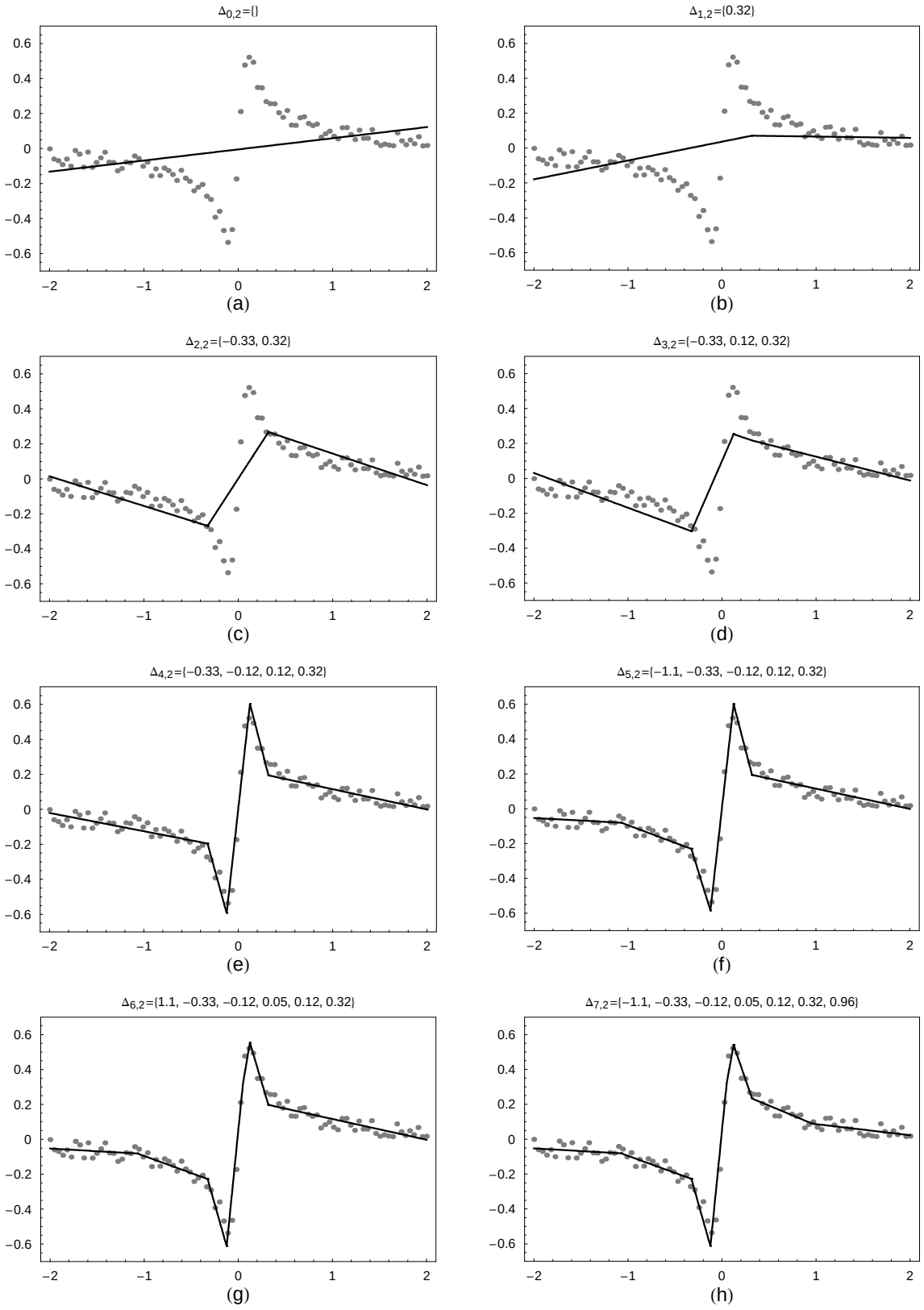

Fig. 4. (Example 1) Graphs of the linear B-spline fits, produced on each consecutive iteration on Stage A of GeDS, except the final one.

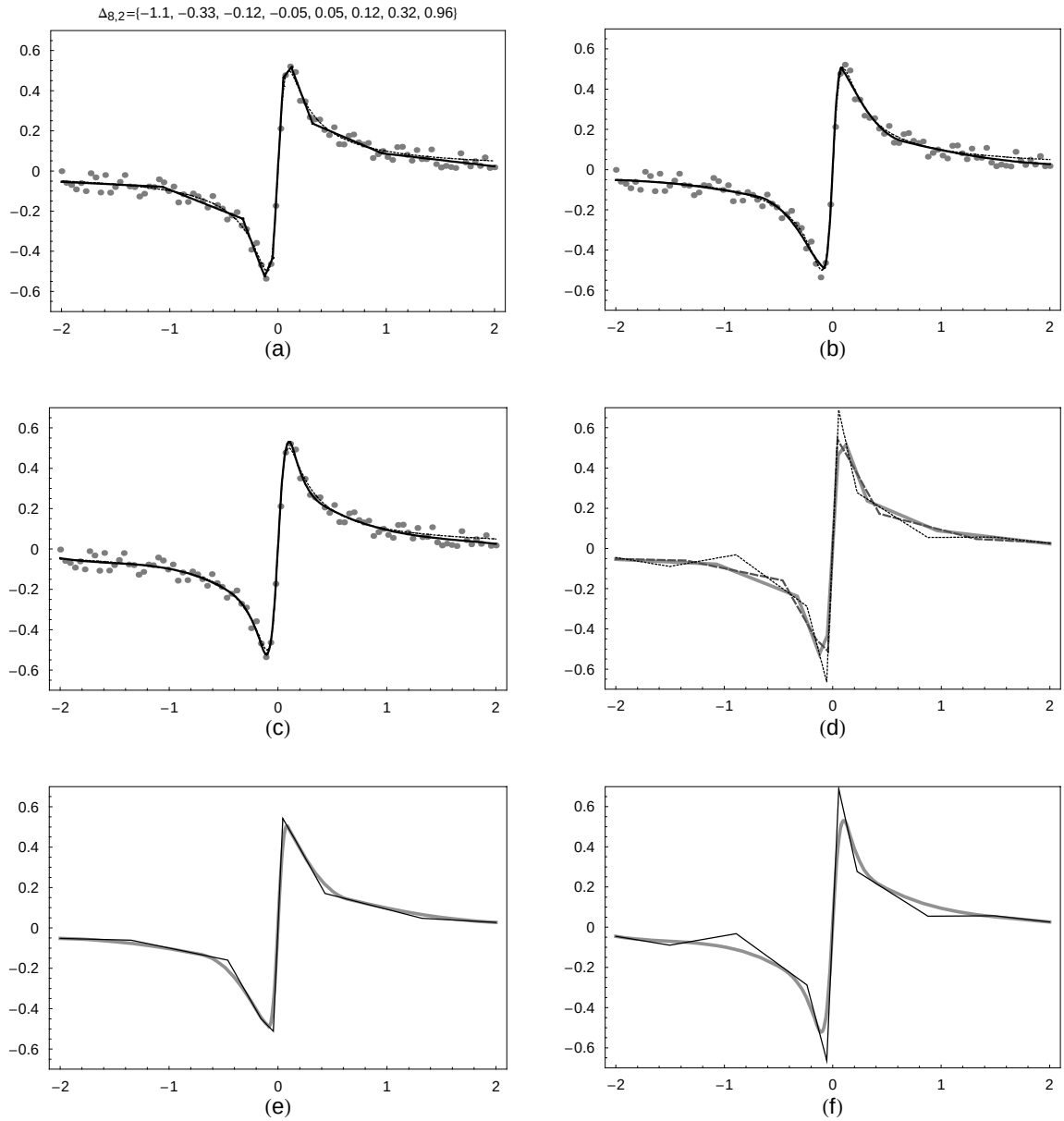

Fig. 5. (Example 1) Graphs of the final B-spline fits, produced by GeDS: (a) linear; (b) and (e) quadratic, correspondingly without and with its control polygon; (c) and (f) cubic, correspondingly without and with its control polygon; (d) the control polygons of the fits in (a) - the thick line, in (b) - the dashed line and in (c) - the dotted line; The dotted function in (a), (b), (c) is the true function.

The details of the final linear, and its corresponding quadratic and cubic spline fits for Example 1 are presented in Table 3. Note that the values for α_{exit} and β are the "automatic" preassigned values $0.9, 0.5$. As can be seen, the function f_1 is symmetric and GeDS places, symmetrically around the origin, 8, 7 and 6 knots, respectively for the linear, quadratic and cubic fits. As seen from Table 3, all the fits are of a very good quality with respect to the MSE. The 400 linear, quadratic and cubic GeD spline fits,

with median number of regression functions $n+k=10$, have median L_2-errors correspondingly 0.26, 0.267, 0.264, which are lower than 0.277, obtained by Schwetlick and Schütze (1995) for a quartic fit with the same number of regression parameters and optimally located knots. For all 400 linear fits, the number of internal knots used by GeDS is 8 or 9. Let us note that the computation time for the fits given in Table 3 is less then a second (0.89 sec.) and does not involve any complicated search procedures. We have produced also a quartic GeDS fit which has five internal knots as does the optimal quartic fit of Schwetlick and Schütze (1995), obtained starting from fifteen knots and after three time consuming knot generation, removal and relocation stages. Our quartic fit has L_2-error equal to 0.46 which indicates that it does not deviate considerably from the (locally) optimal solution.

Table 3. (Example 1) The linear, and its corresponding quadratic and cubic fits produced by GeDS .

Fit No	***Graph***	***n***	***k***	***Internal knots***	α_{exit}, β	L_2 ***– error, MSE***
1	Fig. 5, (a)	2	8	{−1.1, −0.33, −0.12, −0.05, 0.05, 0.12, 0.32, 0.96}	0.9, 0.5	0.2699, 0.000189
2	Fig. 5, (b)	3	7	{−0.69, −0.22, −0.09, 0.00, 0.09, 0.22, 0.64}	0.9, 0.5	0.2944, 0.000127
3	Fig. 5, (c)	4	6	{−0.51, −0.17, −0.04, 0.04, 0.16, 0.47}	0.9, 0.5	0.2631, 0.000119

Example 2. This smooth function first appears as a test example in Fan and Gijbels (1995). It has been used later by Luo and Wahba (1997), Denison at al. (1998) and Zhou and Shen (2001) to test their fitting procedures. With this example, we illustrate that our algorithm works well for data sets with different sample sizes and various noise levels, assuming ϵ is normally distributed. It takes between 0.89 sec and 1.66 sec to compute the GeDS fits, given in Table 4. The L_2-errors of all the fits are within the noise level and their visual quality is very good, as can be seen from Fig. 6. The median MSE value of the 400 linear fits, for $\sigma_\epsilon = 0.4$, with median number of internal knots $k=5$, is 0.009. This is lower than the MSE value 0.012 of Luo and Wahba (1997), and is equal to that of Zhou and Shen (2001), both obtained using cubic splines with higher number of regression functions (e.g., 13 for the fit of Luo and Wahba (1997)). Let us note that for all 400 linear fits the number of internal knots used by GeDS is between 4 and 6. The linear and cubic fits corresponding to the quadratic fit No 3, Table 4, have five and three internal knots and 0.0066 and 0.0277 MSE values respectively.

Table 4. (Example 2) Summary of fits produced by GeDS.

Fit No	***Graph***	N	σ_ϵ	***n***	***k***	***Internal knots***	α_{exit}, β	L_2 ***– error, MSE***
1	Fig. 6, (a)	150	0.25	3	4	{0.37, 0.46, 0.54, 0.62}	0.9, 0.5	2.87, 0.001282
2	Fig. 6, (b)	256	0.25	3	4	{0.38, 0.46, 0.54, 0.63}	0.9, 0.5	4.01, 0.001359
3	Fig. 6, (c)	256	0.4	3	4	{0.38, 0.46, 0.54, 0.60}	0.95, 0.5	6.17, 0.006573
4	Fig. 6, (d)	256	0.6	3	5	{0.26, 0.39, 0.51, 0.55, 0.62}	0.95, 0.5	9.03, 0.021918

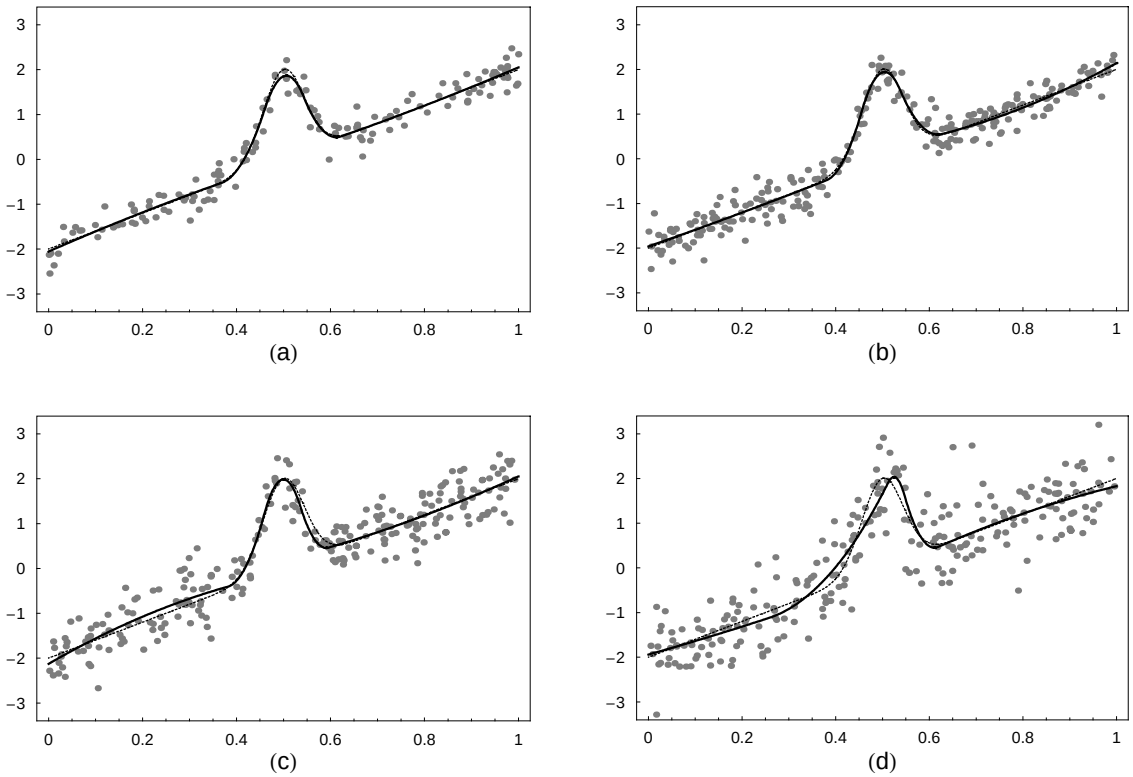

Fig. 6. (Example 2) Graphs of the final quadratic B-spline fits, produced by GeDS: (a) $N = 150$, $\sigma = 0.25$; (b) $N = 256$, $\sigma = 0.25$; (c) $N = 256$, $\sigma = 0.4$; (d) $N = 256$, $\sigma = 0.6$; The dotted function in (a), (b), (c), (d) is the true function.

Note that the first two fits in Table 4 are obtained with $\alpha_{\text{exit}} = 0.9$ and $\beta = 0.5$. Since the noise levels for fits No 3 and 4 are higher than for fits No 1 and 2, α_{exit} has been increased to 0.95, because, in the case of a smooth function and a high noise level, the relative improvements in RSS from one step to another would be smaller and more steps would be needed to recover the function.

Example 3. The function f_3 (see Table 1) appears as a test example in Fan and Gijbels (1995), Luo and Wahba (1997), Denison at al. (1998) and Zhou and Shen (2001). Using the GeDS algorithm we have produced linear, quadratic and cubic fits whose details are given in Table 5. The SNR of the sample data is 3, as for fit No 3 of Example 2. Since f_3 is also relatively smooth we have used $\alpha_{\text{exit}} = 0.95$ and $\beta = 0.5$ in order to obtain the fits in Fig. 7 (e) and (f), which have very good visual quality and low MSE values. The GeD spline fits No 1-3 of Table 5, which have the same number of regression functions $k + n$, are obtained with the preassigned "automatic" values $\alpha_{\text{exit}} = 0.9$ and $\beta = 0.5$. As seen from Fig. 7 (a), (c) and (d), the linear and quadratic fits are sufficiently accurate while the cubic one underfits the data. Adding one more knot by running GeDS with the higher value of $\alpha_{\text{exit}} = 0.95$ improves the cubic fit as illustrated by Fig. 7 (f). The behavior of the stopping rule is illustrated in Fig.7 (b). It can be seen that with $\alpha_{\text{exit}} = 0.9$ the algorithm exits with 6 internal knots for the linear fit and the RSS/N is 0.082677. This

means that the MSE of the linear fit with 8 knots is at least 90% of the value 0.082677, i.e., the MSE has stabilized for three consecutive steps at which models with 6, 7 and 8 knots have been computed. If $\alpha_{\text{exit}} = 0.95$ the algorithm exits one step later, with 7 internal knots for the linear fit and $\text{RSS}/N = 0.07967$ since the improvement in RSS/N for the next two consecutive steps is less than 5% of 0.07967. So, we see that our stopping rule, based on the idea of exiting upon reaching a certain level of stabilization in MSE, selects models with the appropriate number of knots.

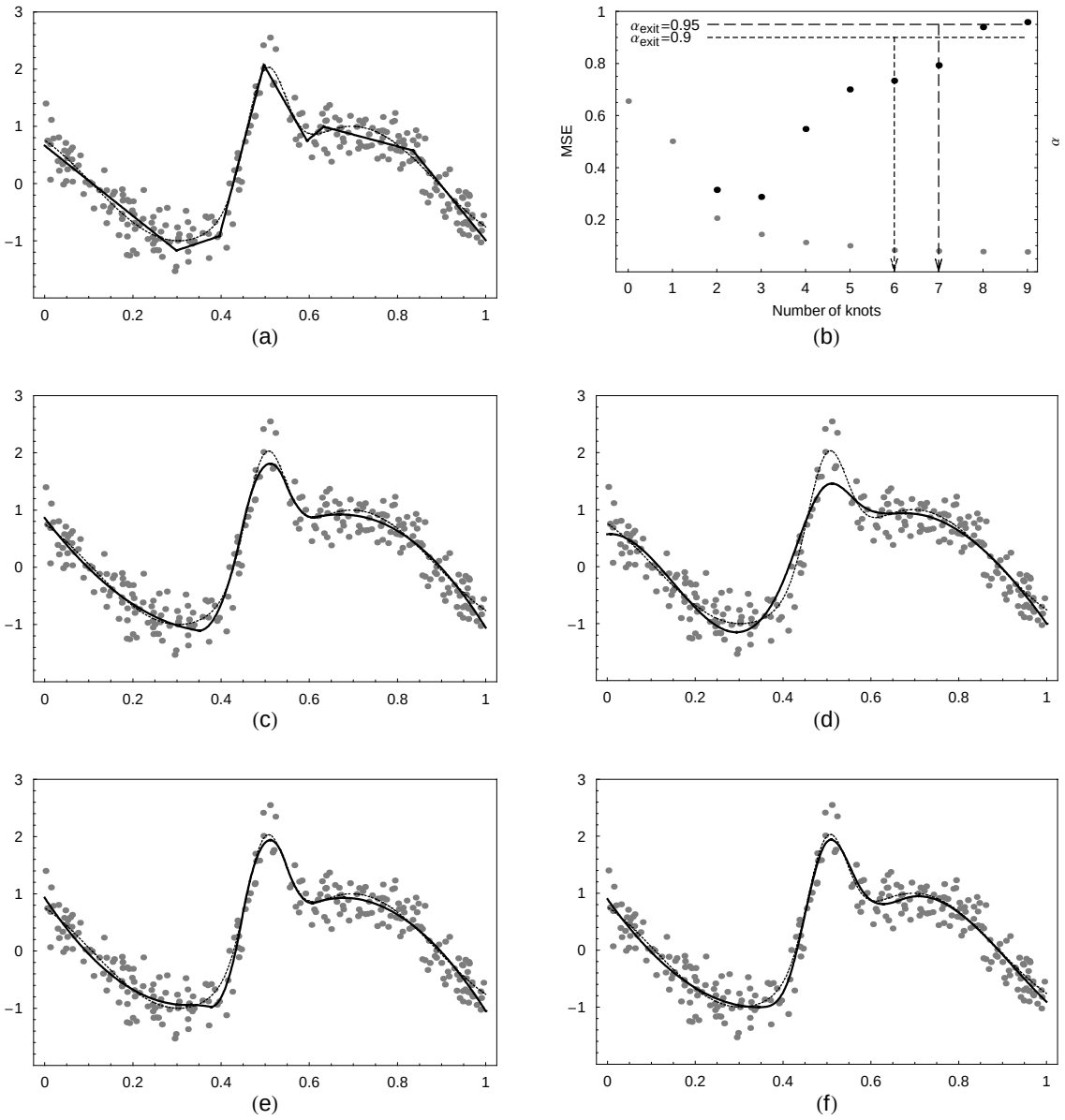

Fig. 7. (Example 3) Graphs of the final B-spline fits, produced by GeDS: (a) linear; (c) and (e) quadratic; (d) and (f) cubic; (b) for each step of the fitting process the values of: α-ratio - black dots, MSE - grey dots; The dotted function in (a), (c), (d), (e), (f) is the true function.

Table 5. (Example 3) Summary of fits produced by GeDS for Example 3.

Fit No	***Graph***	***n***	***k***	***Internal knots***	α_{exit}, β	L_2 **−** ***error, MSE***
1	Fig. 7, (a)	2	6	{0.30, 0.40, 0.50, 0.60, 0.63, 0.83}	0.9, 0.5	4.60, 0.009931
2	Fig. 7, (c)	3	5	{0.35, 0.45, 0.55, 0.61, 0.73}	0.9, 0.5	4.63, 0.005961
3	Fig. 7, (d)	4	4	{0.40, 0.50, 0.57, 0.69}	0.9, 0.5	4.99, 0.019523
4	Fig. 7, (e)	3	6	{0.33, 0.37, 0.45, 0.55, 0.61, 0.73}	0.95, 0.5	4.53, 0.006153
5	Fig. 7, (f)	4	5	{0.35, 0.42, 0.50, 0.57, 0.69}	0.95, 0.5	4.51, 0.004258

The median MSE value for the 400 linear and quadratic fits are respectively equal to 0.0075 and 0.0095, and are comparable with those produced by other authors. For example, Luo and Wahba (1997) report MSE = 0.007 and number of basis functions equal to 13 for their HAS models. For all 400 linear fits the number of internal knots used by GeDS is between 5 and 7. It takes 1.58 sec to compute fits No 1-3 and 1.88 sec to compute fits No 4 and 5 of Table 5.

Example 4. The HeaviSine function is one of the four functions introduced by Donoho and Johnstone (1994) and widely used as test examples by other authors, e.g. Fan and Gijbels (1995), Luo and Wahba (1997), Denison at al. (1998), Zhou and Shen (2001), Lee (2000), Pittman (2002). It is a smooth function with two discontinuities at $x = 0.3$ and $x = 0.72$. It takes 55 seconds to obtain simultaneously the linear, quadratic and cubic GeD spline fits, illustrated in Fig. 8. Their details are given in Table 6. In this and the following examples of spatially inhomogeneous curves, we have set the value for α_{exit} at 0.99, to prevent GeDS from producing a spline approximation which is too smooth for adequately representing the "shape" of the data.

Table 6. (Example 4) Summary of fits produced by GeDS for Example 4.

Fit No	***Graph***	***n***	***k***	***Internal knots***	α_{exit}, β	L_2 **−** ***error MSE***
1	Fig. 8, (a)	2	18	{0.10, 0.13, 0.18, 0.29, 0.30, 0.30, 0.32, 0.38, 0.44, 0.57, 0.63, 0.71, 0.71, 0.72, 0.74, 0.83, 0.84, 0.99}	0.99, 0.5	46.56 0.2203
2	Fig. 8, (b)	3	17	{0.11, 0.16, 0.23, 0.29, 0.30, 0.31, 0.35, 0.41, 0.50, 0.60, 0.67, 0.71, 0.72, 0.73, 0.79, 0.84, 0.92}	0.99, 0.5	43.42 0.0482
3	Fig. 8, (c)	4	16	{0.14, 0.20, 0.26, 0.30, 0.31, 0.33, 0.38, 0.46, 0.55, 0.64, 0.69, 0.72, 0.73, 0.77, 0.81, 0.89}	0.99, 0.5	44.82 0.0942

For the quadratic GeDS fit for example 4, the median number of regression functions $k + n$ is only 20 while the median MSE value 0.057, is comparable with 0.04 given by Luo and Wahba (1997) for their cubic spline model with 50 basis functions. Our GeDS algorithm uses between 17 and 21 internal knots to fit the 31 simulated data sets in the linear case. Based on the L_2-errors for the linear, quadratic and cubic fits given in Table 6, one can see that the best fit for this particular function is of degree 2.

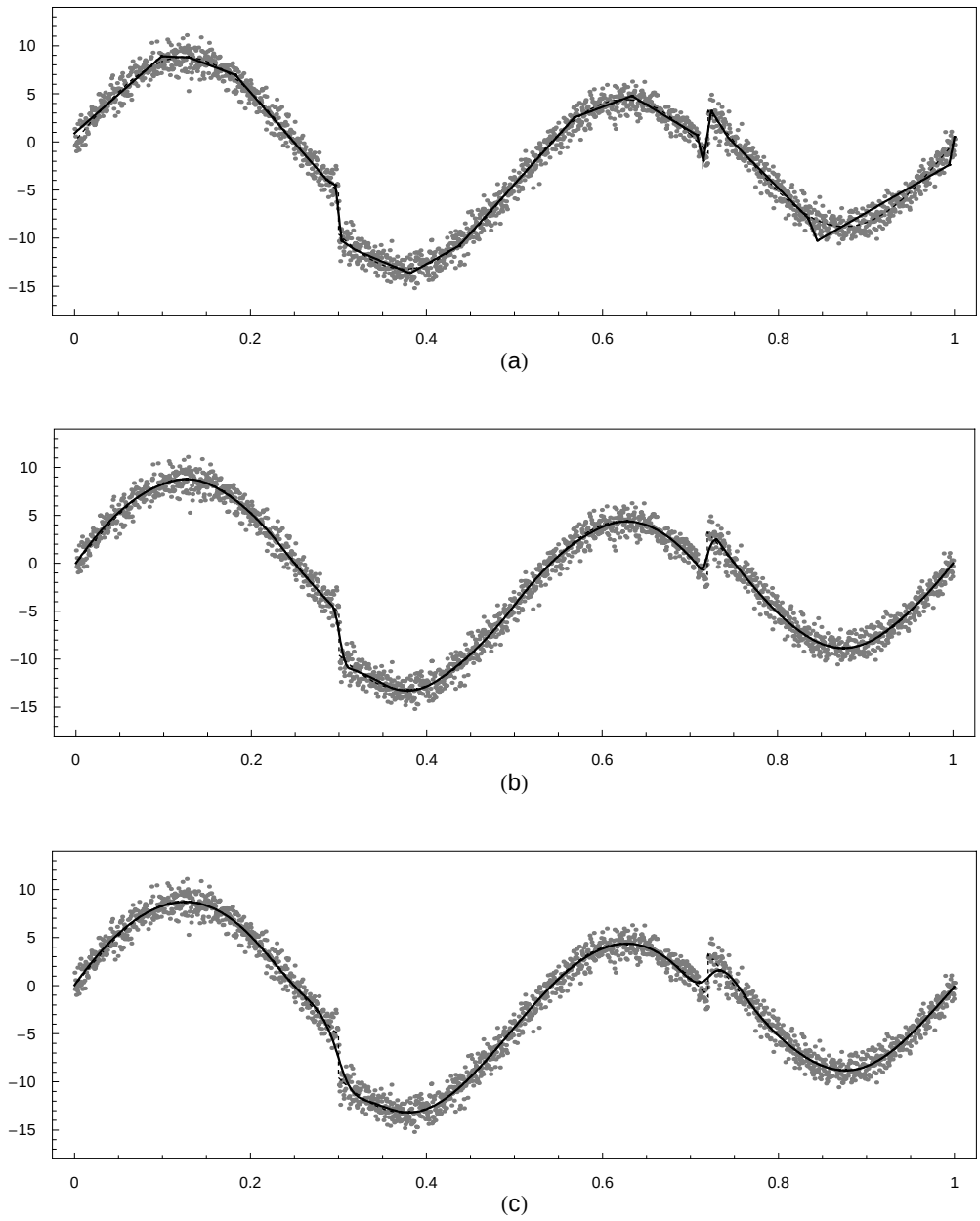

Fig. 8. (Example 4) Graphs of the final B-spline fits, produced by GeDS: (a) linear; (b) quadratic; (c) cubic; The dotted function in (a), (b), (c) is the true function.

Example 5. This, function is known as the Doppler function. It is highly oscillating, especially around the origin and is difficult to recover. Using the GeDS algorithm we have obtained six different fits for the same data set with SNR equal to 7. Fits No 1-3, given in Table 7, are calculated simultaneously in 304 seconds with $\alpha_{\text{exit}} = 0.99$. The quadratic one has 46 knots and MSE = 0.13. For comparison the HAS cubic fit, produced by Luo and Wahba (1997) has MSE = 0.10 with 120 basis functions. Based on the quadratic GeD spline fits of 31 simulated data sets we have obtained median MSE

value 0.089 and median number of knots 62, using $\alpha_{\text{exit}} = 0.999$. The number of knots for the 31 quadratic fits was between 50 and 78.

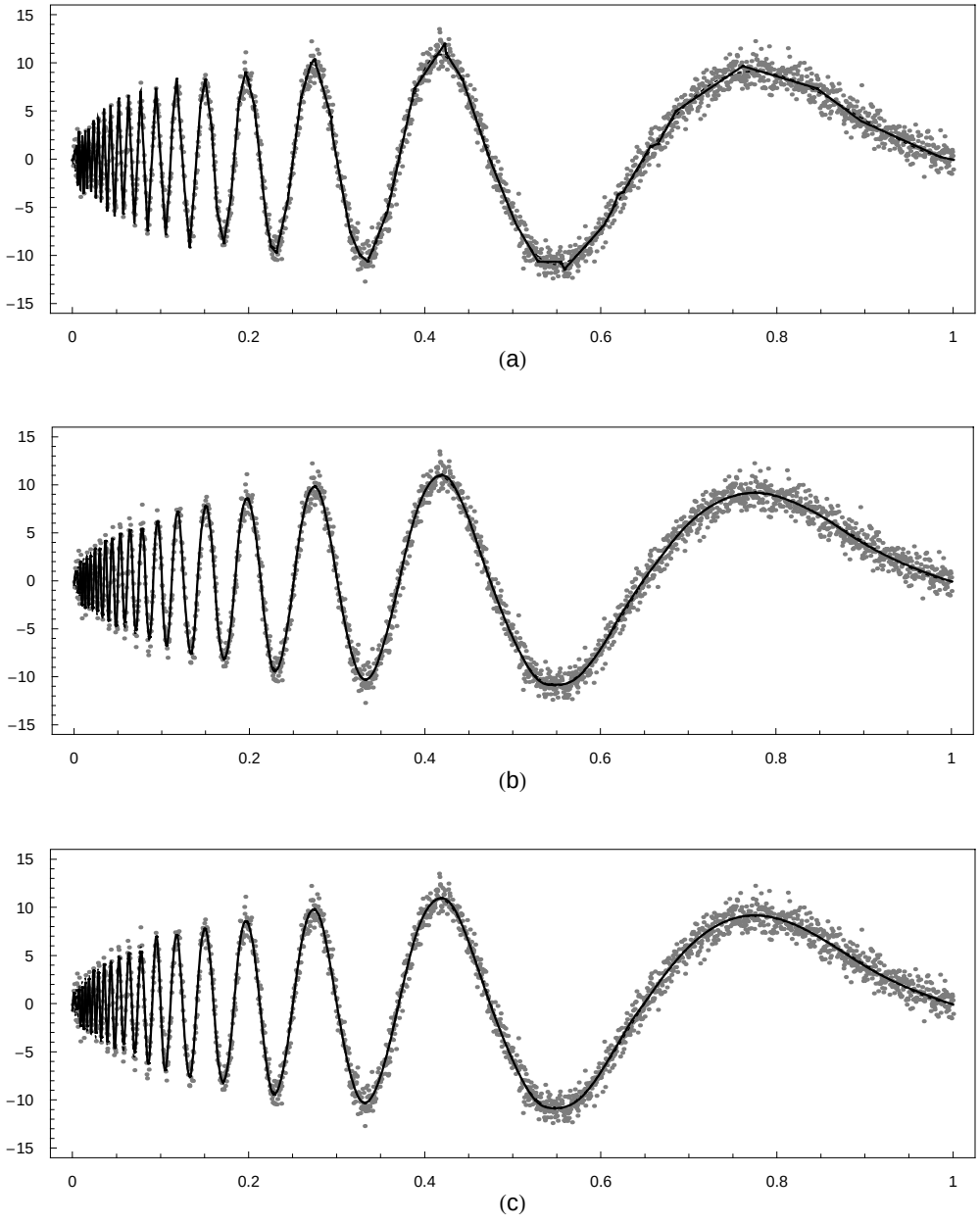

Fig. 9. (Example 5) Graphs of the final B-spline fits, produced by GeDS: (a) linear; (b) quadratic; (c) cubic; The dotted function in (a), (b), (c) is the true function.

Comparing the L_2-errors of the fits of degree 1,2 and 3, summarized in Table 7 the best fit for the Doppler function is the quadratic one. The GeDS fits No 4-6, given in Fig. 9 are obtained in 717 seconds.

Table 7. (Example 5) Summary of fits produced by GeDS for Example 5.

Fit No	***Graph***	***n***	***k***	α_{exit}, β	L_2 **– error, MSE**
1	–	2	47	0.99, 0.5	48.24, 0.199802
2	–	3	46	0.99, 0.5	46.77, 0.125328
3	–	4	45	0.99, 0.5	49.04, 0.233945
4	Fig. 9, (a)	2	74	0.999, 0.5	45.21, 0.114633
5	Fig. 9, (b)	3	73	0.999, 0.5	44.92, 0.060037
6	Fig. 9, (c)	4	72	0.999, 0.5	46.10, 0.106811

Example 6. The Bumps function is very wiggly and difficult to fit as well. Following the prescription for choosing α_{exit} in the case of fitting wiggly functions with high SNR, we have set $\alpha_{\text{exit}} = 0.99$ and have obtained the GeDS fits whose details are summarized in Table 8. Looking at the L_2-errors we see that the fit with the lowest L_2-error is the linear one. A linear fit for Bumps is given also by Lee (2000) whose MDL procedure automatically chooses the order of the fit within the range 1 to 4. Based on 31 simulated data sets the median MSE value for the linear fit is 0.22, for the quadratic fit is 0.51 and the median number of knots is 90. The GeDS algorithm places between 79 and 102 knots for these 31 fits. For comparison the median MSE value reported by Pittman (1994) for the cubic AGS fit is 0.4001, for a certain median number of knots, which is not given.

Table 8. (Example 6) Summary of fits produced by GeDS for Example 6.

Fit No	***Graph***	***n***	***k***	α_{exit}, β	L_2 **– error, MSE**
1	–	2	83	0.99, 0.5	48.59, 0.283631
2	–	3	82	0.99, 0.5	56.03, 0.631448
3	–	4	81	0.99, 0.5	66.44, 1.198390
4	Fig. 10, (a)	2	103	0.999, 0.5	44.51, 0.140580
5	Fig. 10, (b)	3	102	0.999, 0.5	47.96, 0.264664
6	Fig. 10, (c)	4	101	0.999, 0.5	52.29, 0.445403

The fits No 1-3 are obtained in 795 seconds, whereas fits No 4-6, given in Fig. 10 are computed in 1255 seconds. Let us recall that, due to their shape preserving property, higher order B-spline curves deviate more strongly from their control polygons, which explains why the quadratic and cubic B-spline fits in Fig. 10 (b) and (c) reproduce the very narrow and high spikes of the Bumps function less well than the linear fit in Fig. 10 (a).

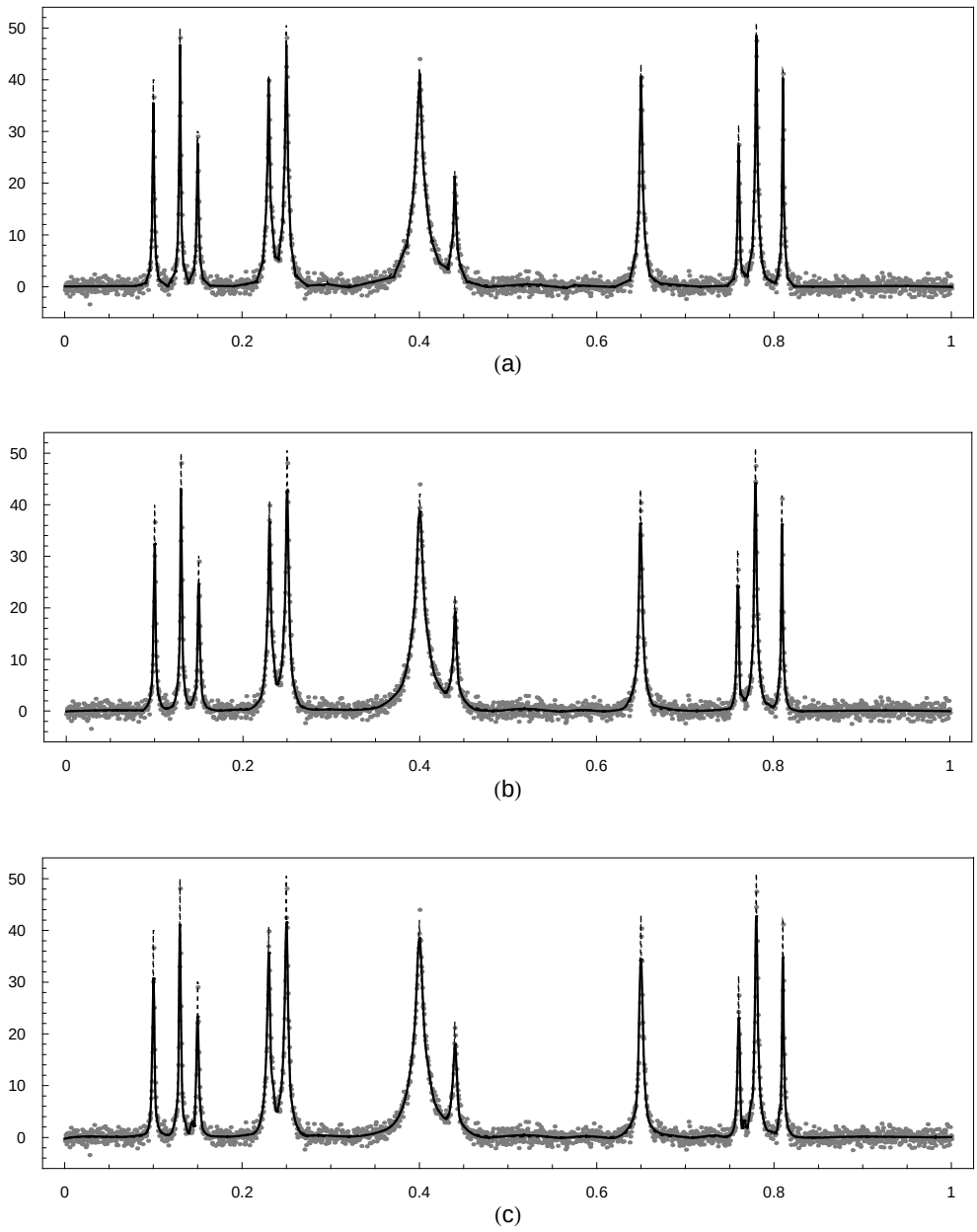

Fig. 10. (Example 6) Graphs of the final B-spline fits, produced by GeDS: (a) linear; (b) quadratic; (c) cubic; The dotted function in (a), (b), (c) is the true function.

Example 7. For the Blocks function, in order to obtain fits No 1-4 given in Table 9, we have run our algorithm with $\alpha_{\text{exit}} = 0.99$ and $\alpha_{\text{exit}} = 0.999$. We have obtained linear, quadratic and cubic fits for this example, but only results for the linear and quadratic cases are presented in Fig 11, since the MSE increases with the degree, i.e., as seen from Table 9, the L_2-error of the linear fit is lowest. Let us note once again that, since the GeDS algorithm produces simultaneously fits of different order, it may be considered as a free parameter. The order of the fit, with lowest L_2-error, may serve as an estimate for

this free parameter which is estimated by Lee (2000) applying his time consuming MDL optimization procedure.

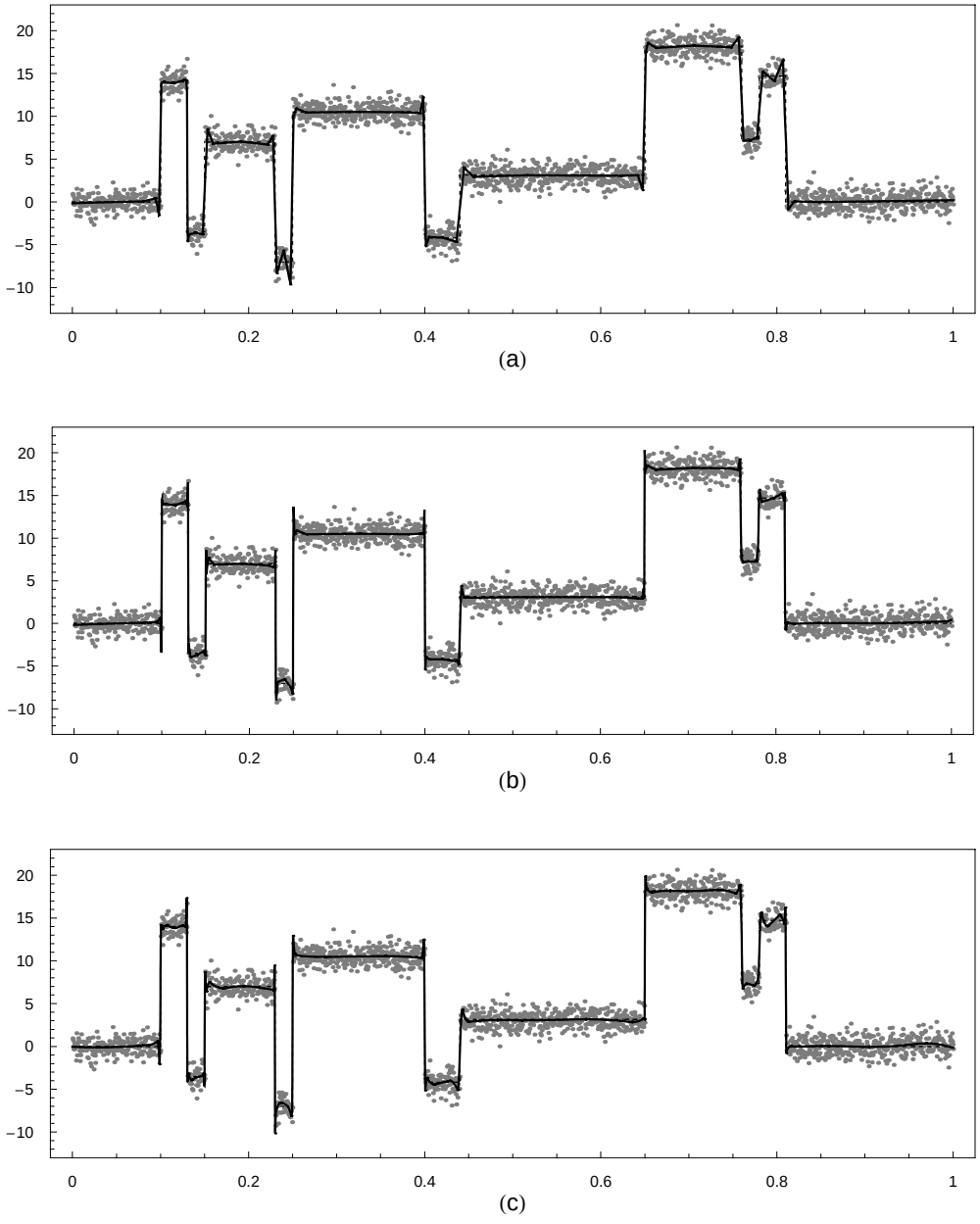

Fig. 11. (Example 7) Graphs of the final B-spline fits, produced by GeDS: (a) linear; (b) linear; (c) quadratic; The dotted function in (a), (b), (c) is the true function.

The fits No 1-2 are obtained in 344 seconds and No 3-4 in 856 seconds. Our median MSE value, based on 31 runs with $\alpha_{\text{exit}} = 0.999$ is 0.12 with 83 median number of knots. For comparison, the median MSE value given by Zhou and Shen (2001) is 0.08, who do not report the number of knots.

Table 9. (Example 7) Summary of fits produced by GeDS for Example 7.

Fit No	***Graph***	***n***	***k***	α_{exit}, β	L_2 **– error, *MSE***
1	Fig. 11, (a)	2	53	0.99, 0.5	55.63, 0.642906
2	–	3	52	0.99, 0.5	59.80, 0.860989
3	Fig. 11, (b)	2	85	0.999, 0.5	42.43, 0.082962
4	Fig. 11, (c)	3	84	0.999, 0.5	43.68, 0.126953

Example 8. The Titanium Heat Data example, was first considered by DeBoor (1968) (see also DeBoor 2001), and used as a test example by Jupp (1978), Hu (1993) and Schwetlick and Schütze (1995). It is suitable for comparing variable knot spline algorithms, since the real data, and the spline fits with corresponding knots and L_2-errors have been published. As a result of running GeDS we have obtained linear, quadratic and cubic spline fits, illustrated in Fig. 12 and 13. The linear fit with 6 knots, obtained after stage A of GeDS in 0.49 seconds with $\alpha_{\text{exit}} = 0.9$ and $\beta = 0.5$, is given in Fig. 12 (a). Its corresponding quadratic spline fit with 5 internal knots, obtained after stage B, of GeDS is given in Fig. 12 (b).

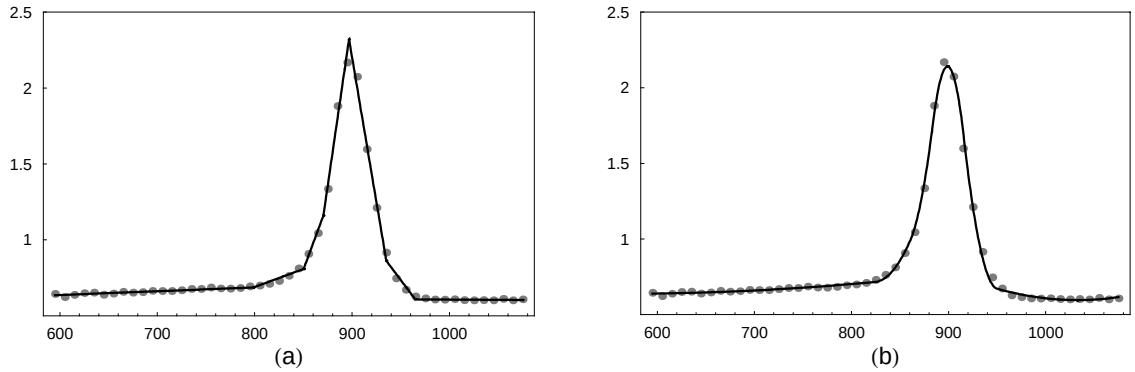

Fig. 12. (Example 8) The linear (a) and quadratic (b) B-spline fits to the Titanium Heat data, produced by GeDS.

Changing the values of α_{exit} and β to 0.8 and 0.6 respectively, leads to the GeDS fits, given in Fig. 13. They have lower L_2-error than the fits No 1 and 2, (see Table 10), and number of internal knots, is respectively 11 for the quadratic and 10 for the cubic models.

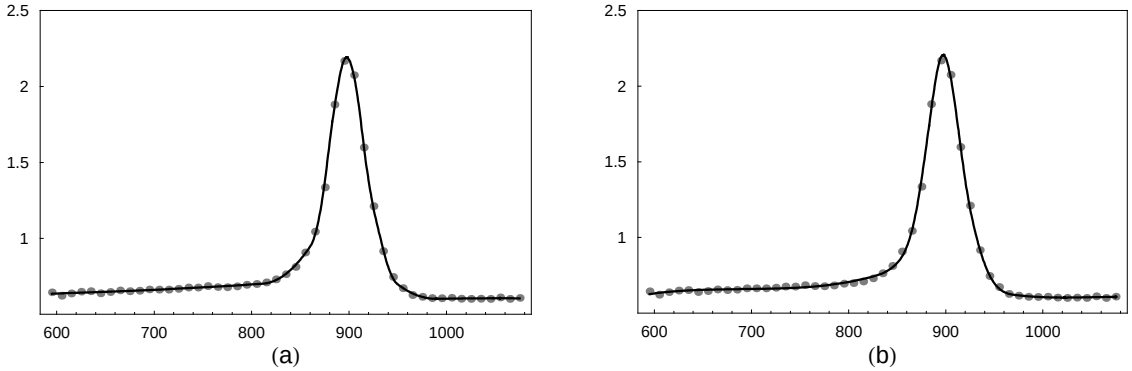

Fig. 13. (Example 8) The quadratic (a) and cubic (b) B-spline fits to the Titanium Heat data, produced by GeDS.

Table 10. (Example 8) Summary of fits produced by GeDS for Example 8.

Fit No	***Graph***	***n***	***k***	***Internal knots***	α_{exit}, β	L_2 ***− error***
1	Fig. 12, (a)	2	6	{798.61, 850.23, 870.49, 896.79, 935.07, 964.77}	0.9, 0.5	0.1606
2	Fig. 12, (b)	3	5	{824.42, 860.36, 883.64, 915.93, 949.92}	0.9, 0.5	0.1695
3	Fig. 13, (a)	3	11	{811.18, 836.99, 860.36, 877.74, 890.90, 900.90, 912.52, 927.52, 935.03, 949.92, 990.01}	0.8, 0.6	0.0617
4	Fig. 13, (b)	4	10	{824.20, 848.16, 868.57, 884.09, 895.60, 907.28, 920.01, 930.03, 944.95, 971.69}	0.8, 0.6	0.0919

For comparison, the cubic spline fit of DeBoor and Rice (1968) has L_2-error equal to 0.092. Jupp (1978) has found the optimum knot location of a cubic spline fit with five internal knots, for which the L_2-error is equal to 0.087. We have used, the internal knots of fit No 2 (see Table 10), produced after Stage B of GeDS algorithm, as an initial approximation for a Fibonacci knot optimization search in order to get some improvement in the L_2-error value. As a result, we have obtained a quadratic B-spline fit with 5, slightly adjusted, internal knots, {817.82, 863.33, 882.38, 909.49, 955.23}, and with L_2-error = 0.0545 which turns out to be lower than the optimal value 0.087 of Jupp (1978) for the cubic case.

7. Discussion and comparison with other methods

Based on the results of Section 6, we can conclude that the GeDS algorithm can be used successfully for fitting both smooth and spatially inhomogeneous functions. It is a fast, stable, non-complicated algorithm with an appropriate geometric interpretation which allows the user to follow the entire fitting process. The algorithm is automatic, in the sense that one can use the default values of the two input parameters α_{exit} and β to successfully fit relatively smooth functions, and fit less smooth functions applying another pair of values for α_{exit} and β. Thus, automation is combined with some flexibility in controlling the output, which is often more convenient than the entirely automatic approach. The existence of two input parameters gives flexibility in tuning GeDS to cope with the particular noise level and smoothness characteristics of the underlying function. The results of fitting the smooth functions in Section 6 show that the spline models, produced applying the default values $\alpha_{\text{exit}} = 0.9$, $\beta = 0.5$, are comparable with those obtained with other methods.

There are no restrictions for GeDS with respect to the data set $\{x_i, y_i\}_{i=1}^{N}$, in contrast to some of the other algorithms. Thus, the wavelet shrinkage method of Donoho and Johnstone (1994) requires the x values to be equally spaced and N to be a power of 2. Other methods, e.g. SARS, HAS and the method of automatic Bayesian curve fitting of Denison at al. (1998) seem to require rescaling of the function to the [0, 1] interval.

Our algorithm does not require any initial guess for the possible number and/or position of the knots in contrast to, other methods e.g., NEWKNOT of DeBoor and Rice (1968), Schwetlick and Schütze (1995), Zhou and Shen (2001). Although Schwetlick and Schütze (1995) give some recommendations for the number of knots, their results are seen to be sensitive with respect to this parameter. The Bayesian algorithm of Denison at al. (1998) requires a guess for the prior number of knots and it chooses the candidate knot locations among the N regular grid points on the range of x. The AGS method also chooses the possible knot sites among the abscissae values of the data points. The GeDS algorithm does not restrict the minimum and maximum possible number of knots as do, e.g., HAS and AGS methods.

One of the most important characteristics of the GeDS algorithm is that it gives simultaneously linear, quadratic, cubic, etc. fits because once the LS linear B-spline fit on Stage A is found, using the averaging knot location method (13), one immediately obtains the knots for the higher order B-spline fits. As far as we have been able to establish, no other spline fitting procedure is capable of doing this. Hence, one has the flexibility to choose the degree of the fit providing best compromise between smoothness and accuracy.

As an alternative to the stopping rule (24) we have implemented two additional stopping methods according to which our algorithm exits with the number of knots which minimizes Stein's unbiased risk estimate (SURE) (see Stein (1981))

$$R(\hat{f}) = \left\{ \sum_{i=1}^{N} \left(y_i - \hat{f}_{\Delta_{k,n}}(x_i) \right)^2 \right\} \Big/ N + D \, \frac{(k+n-1)}{N} \, \sigma^2 \qquad (25)$$

or the generalized cross validation (GCV) (see e.g., Craven and Wahba (1979))

$$\mathrm{GCV}(\hat{f}) = \left(\frac{\sum_{i=1}^{N} \left(y_i - \hat{f}_{\Delta_{k,n}}(x_i) \right)^2}{N} \right) \Big/ \left(1 - \frac{d(k)}{N} \right)^2 \qquad (26)$$

criterion. We have assumed that the minimum is attained when SURE or GCV do not decrease in two consecutive steps. Rules (25) and (26) depend on the choice of the parameters D and $d(k)$, and when $D = 2$ and $d(k) = k + 1$ they behave roughly as (24), although in some of the cases, e.g., in Examples 3 and 6 they force GeDS to exit much later than (24), causing some overfit. The choice $D = 3$ and $d(k) = 3\,k + 1$, as noted by Zhou and Shen (2001) tends to yield a smaller model, underfitting the underlying function. For a comparative study of different model selection methods, we refer to Lee (2002).

As a general conclusion we believe that, with (25) and (26), GeDS becomes entirely automatic and can be applied if such a feature is preferred to the flexibility of controlling the output provided by rule (24).

In conclusion, we have demonstrated in this paper that the proposed GeDS method provides a novel approach to choosing the number and position of the knots. It produces simultaneously, linear quadratic, cubic and higher degree fits. It is motivated by geometrical arguments and does not place knots through complex stochastic or deterministic optimization of GCV, SURE, MDL or any other function of RSS or/and $k + n$ within a multivariate parameter space. However, it provides stepwise minimization of the RSS as seen from Fig. 7 (b) which is terminated following an appropriate stopping rule, involving an adjustable exit parameter. Another positive characteristic is that GeDS can be extended to multivariate non-parametric smoothing. Details of how this may be done are outside the scope of this paper and are subject of ongoing research.

Acknowledgements

The authors would like to acknowledge support received by the UK Institute of Actuaries.

References

Biller, C. (2000). Adaptive Bayesian regression splines in semiparametric generalized linear models. *J. Comput. and Graph. Stat.*, **9**, 122-140.

Cohen, E., Riesenfeld, R. F. and Elber, G. (2001). *Geometric Modelling with Splines: An Introduction.* , Natick, Massachusetts: A K Peters.

Cox, M., Harris, P. and Kenward, P. (2002). Fixed and free-knot univariate least-squares data approximation by polynomial splines. *Technical Report CMSC 13/02, National Physical Laboratory, Teddington, UK.*

Craven, P. and Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the generalized cross validation. *Numerische Mathematik*, **31**, 337-403.

De Boor, C. (2001). *A practical Guide to Splines*, Revised Edition, New York: Springer.

De Boor, C. and Rice, J. (1968). Least squares cubic spline approximation II. Variable knots. Comp. Sci.Dpt. *Technical Report* 21, Purdue Univrsity, West Laffayet, Indiana.

Denison, D., Mallick, B., and Smith, A. (1998). Automatic Bayesian curve fitting, *J. R. Statist. Soc.*, B, **60**, 333-350.

Donoho, D. and Johnstone, I (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, **81**, 425-455.

———— (1995). Adapting to unknown smoothness via wavelet shrinkage. *J. Am. Statist. Ass.*, **90**, 1200-1224.

Eubank, R.(1988). *Spline smoothing and Nonparametric Regression.* Dekker, New York.

Fan, J. and Gijbels, I. (1995). Data-driven bandwidth selection in local polynomial fitting:Variable bandwidth and spatial adaptation. *J. R. Statist. Soc.*, B, **57**, 371-394.

Farin, G. (2002). *Curves and Surfaces for CAGD*, Fifth Edition, San Francisco: Morgan Kaufmann.

Friedman, J. H. (1991). Multivariate adaptive regression splines (with discussion). *Ann. Statist*. **19**, 1-141.

Friedman, J. H. and Silverman, B.W. (1989). Flexible Parsimonious smoothing and additive modeling (with discussion).*Technometrics*., **31**, . 3-39.

Hu, Y. (1993). An algorithm for data reduction using splines with free knots. *IMA J. Numer. Anal.*, **13**, 328-343.

Jupp, D. (1978). Approximation to data by splines with free knots. *SIAM J. Num. Analysis.*, **15**, 328-343.

Kaishev, V. K. (1984). A computer program package for solving spline regression problems, In: *Proceedings in Computational Statistics, COMPSTAT* (eds T. Havranek, Z. Sidak and M. Novak), pp. 409-414, Wien: Physica-Verlag.

Lee, T. C.M. (2000). Regression spline smoothing using the minimum description length principle. *Stat. & Prob. Letters,* **48**, 71-82.

Lee, T. C.M. (2002). On algorithms for ordinary least squares regression spline fitting: A comparable study. *J. Statist. Comput. Simul.,* **72(8)**, 647-663.

Luo, Z., and Wahba, G. (1997). Hybrid adaptive splines. *J. Am. Statist. Ass.,* **92**, 107-115.

Lytch, T. and Mørken, K. (1993). A data reduction strategy for splines with application to the approximation of functions and data. *IMA J. Numer. Anal.*, **8**, 185-208.

Marx, B. D. and Eilers, P. H.C. (1996). Flexible Smoothing with B-splines and Penalties. *Stat. Science*, **11**, 2, 89-121.

Pittman, J. (2002). Adaptive Splines and Genetic Algorithms. *J. Comput. and Graph. Stat.*, **11**, 3, 1-24.

Rupert, D. (2002). Selecting the number of knots for penalized splines. *J. Comput. and Graph. Stat.,* **11**, 4, 735-757.

Rupert, D., and Carroll, R. J. (2000). Spatially-Adaptive penalties for spline fitting. *Australian and New Zealand Journal of Statistics*, **42**, 205-223.

Schwetlick, H. and Schütze, T. (1995). Least squares approximation by splines with free knots. *BIT. Numerical Math., **35***, 854-866.

Smith, P. L. (1982). Curve fitting and modeling with splines using statistical variable selection techniques. *Report NASA 166034*, Langley Research Center, Hampton, VA.

Smith, M. and Kohn, R. (1996). Nonparametric regression using Bayesian variable selection. *J. Econometrics, **75***, 317-344.

Stein, C. (1981). Estimation of the mean of a multivariate normal. *The Ann. Statist*., **9**, 1135-1151.

Stone, C. J., Hansen, M.H., Kooperberg, C. and Truong, Y. K. (1997). Polynomial Splines and their tensor products in extended linear modeling. *Ann. Statist*., **25**, 1371-1470.

Zhou, S. and Shen, X. (2001). Spatially adaptive regression splines and accurate knot selection schemes. *J. Am. Statist. Ass.,* **96**, 247-259.

Wahba, G. (1990). *Spline Models for Observational Data.* SIAM, Philadelphia

Cass Business School
City of London

FACULTY OF ACTUARIAL SCIENCE AND STATISTICS

Actuarial Research Papers since 2001

135. Renshaw A. E. and Haberman S. On the Forecasting of Mortality Reduction Factors. February 2001.
ISBN 1 901615 56 1

136. Haberman S., Butt Z. & Rickayzen B. D. Multiple State Models, Simulation and Insurer Insolvency. February 2001. 27 pages.
ISBN 1 901615 57 X

137. Khorasanee M.Z. A Cash-Flow Approach to Pension Funding. September 2001. 34 pages.
ISBN 1 901615 58 8

138. England P.D. Addendum to "Analytic and Bootstrap Estimates of Prediction Errors in Claims Reserving". November 2001. 17 pages.
ISBN 1 901615 59 6

139. Verrall R.J. A Bayesian Generalised Linear Model for the Bornhuetter-Ferguson Method of Claims Reserving. November 2001. 10 pages.
ISBN 1 901615 62 6

140. Renshaw A.E. and Haberman. S. Lee-Carter Mortality Forecasting, a Parallel GLM Approach, England and Wales Mortality Projections. January 2002. 38 pages.
ISBN 1 901615 63 4

141. Ballotta L. and Haberman S. Valuation of Guaranteed Annuity Conversion Options. January 2002. 25 pages.
ISBN 1 901615 64 2

142. Butt Z. and Haberman S. Application of Frailty-Based Mortality Models to Insurance Data. April 2002. 65 pages.
ISBN 1 901615 65 0

143. Gerrard R.J. and Glass C.A. Optimal Premium Pricing in Motor Insurance: A Discrete Approximation. (Will be available 2003).

144. Mayhew, L. The Neighbourhood Health Economy. A systematic approach to the examination of health and social risks at neighbourhood level. December 2002. 43 pages.
ISBN 1 901615 66 9

145. Ballotta L. and Haberman S. The Fair Valuation Problem of Guaranteed Annuity Options: The Stochastic Mortality Environment Case. January 2003. 25 pages.
ISBN 1 901615 67 7

146. Haberman S., Ballotta L. and Wang N. Modelling and Valuation of Guarantees in With-Profit and Unitised With-Profit Life Insurance Contracts. February 2003. 26 pages.
ISBN 1 901615 68 5

147. Ignatov Z.G., Kaishev V.K and Krachunov R.S. Optimal Retention Levels, Given the Joint Survival of Cedent and Reinsurer. March 2003. 36 pages.
ISBN 1 901615 69 3

148. Owadally M.I. Efficient Asset Valuation Methods for Pension Plans. March 2003. 20 pages.
ISBN 1 901615 70 7

149. Owadally M.I. Pension Funding and the Actuarial Assumption Concerning Investment Returns. March 2003. 32 pages.

ISBN 1 901615 71 5

150. Dimitrova D, Ignatov Z. and Kaishev V. Finite time Ruin Probabilities for Continuous Claims Severities. Will be available in August 2004.

151. Iyer S. Application of Stochastic Methods in the Valuation of Social Security Pension Schemes. August 2004. 40 pages.

ISBN 1 901615 72 3

152. Ballotta L., Haberman S. and Wang N. Guarantees in with-profit and Unitized with profit Life Insurance Contracts; Fair Valuation Problem in Presence of the Default Option[1]. October 2003. 28 pages.

ISBN 1-901615-73-1

153. Renshaw A. and Haberman. S. Lee-Carter Mortality Forecasting Incorporating Bivariate Time Series. December 2003. 33 pages.

ISBN 1-901615-75-8

154. Cowell R.G., Khuen Y.Y. and Verrall R.J. Modelling Operational Risk with Bayesian Networks. March 2004. 37 pages.

ISBN 1-901615-76-6

155. Gerrard R.G., Haberman S., Hojgaard B. and Vigna E. The Income Drawdown Option: Quadratic Loss. March 2004. 31 pages.

ISBN 1-901615-77-4

156. Karlsson, M., Mayhew L., Plumb R, and Rickayzen B.D**.** An International Comparison of Long-Term Care Arrangements. An Investigation into the Equity, Efficiency and sustainability of the Long-Term Care Systems in Germany, Japan, Sweden, the United Kingdom and the United States. April 2004. 131 pages.

ISBN 1 901615 78 2

157. Ballotta Laura. Alternative Framework for the Fair Valuation of Participating Life Insurance Contracts. June 2004. 33 pages.

ISBN 1-901615-79-0

158. Wang Nan. An Asset Allocation Strategy for a Risk Reserve considering both Risk and Profit. July 2004. 13 pages.

ISBN 1 901615-80-4

Statistical Research Papers

1. Sebastiani P. Some Results on the Derivatives of Matrix Functions. December 1995. 17 Pages.

ISBN 1 874 770 83 2

2. Dawid A.P. and Sebastiani P. Coherent Criteria for Optimal Experimental Design. March 1996. 35 Pages.

ISBN 1 874 770 86 7

3. Sebastiani P. and Wynn H.P. Maximum Entropy Sampling and Optimal Bayesian Experimental Design. March 1996. 22 Pages.

ISBN 1 874 770 87 5

4. Sebastiani P. and Settimi R. A Note on D-optimal Designs for a Logistic Regression Model. May 1996. 12 Pages.

ISBN 1 874 770 92 1

5. Sebastiani P. and Settimi R. First-order Optimal Designs for Non Linear Models. August 1996. 28 Pages.

ISBN 1 874 770 95 6

6. Newby M. A Business Process Approach to Maintenance: Measurement, Decision and Control. September 1996. 12 Pages.

ISBN 1 874 770 96 4

7. Newby M. Moments and Generating Functions for the Absorption Distribution and its Negative Binomial Analogue. September 1996. 16 Pages.
ISBN 1 874 770 97 2

8. Cowell R.G. Mixture Reduction via Predictive Scores. November 1996. 17 Pages.
ISBN 1 874 770 98 0

9. Sebastiani P. and Ramoni M. Robust Parameter Learning in Bayesian Networks with Missing Data. March 1997. 9 Pages.
ISBN 1 901615 00 6

10. Newby M.J. and Coolen F.P.A. Guidelines for Corrective Replacement Based on Low Stochastic Structure Assumptions. March 1997. 9 Pages.
ISBN 1 901615 01 4.

11. Newby M.J. Approximations for the Absorption Distribution and its Negative Binomial Analogue. March 1997. 6 Pages.
ISBN 1 901615 02 2

12. Ramoni M. and Sebastiani P. The Use of Exogenous Knowledge to Learn Bayesian Networks from Incomplete Databases. June 1997. 11 Pages.
ISBN 1 901615 10 3

13. Ramoni M. and Sebastiani P. Learning Bayesian Networks from Incomplete Databases. June 1997. 14 Pages.
ISBN 1 901615 11 1

14. Sebastiani P. and Wynn H.P. Risk Based Optimal Designs. June 1997. 10 Pages.
ISBN 1 901615 13 8

15. Cowell R. Sampling without Replacement in Junction Trees. June 1997. 10 Pages.
ISBN 1 901615 14 6

16. Dagg R.A. and Newby M.J. Optimal Overhaul Intervals with Imperfect Inspection and Repair. July 1997. 11 Pages. ISBN 1 901615 15 4

17. Sebastiani P. and Wynn H.P. Bayesian Experimental Design and Shannon Information. October 1997. 11 Pages. ISBN 1 901615 17 0

18. Wolstenholme L.C. A Characterisation of Phase Type Distributions. November 1997. 11 Pages. ISBN 1 901615 18 9

19. Wolstenholme L.C. A Comparison of Models for Probability of Detection (POD) Curves. December 1997. 23 Pages. ISBN 1 901615 21 9

20. Cowell R.G. Parameter Learning from Incomplete Data Using Maximum Entropy I: Principles. February 1999. 19 Pages. ISBN 1 901615 37 5

21. Cowell R.G. Parameter Learning from Incomplete Data Using Maximum Entropy II: Application to Bayesian Networks. November 1999. 12 Pages ISBN 1 901615 40 5

22. Cowell R.G. FINEX : Forensic Identification by Network Expert Systems. March 2001. 10 pages.
ISBN 1 901615 60X

23. Cowell R.G. When Learning Bayesian Networks from Data, using Conditional Independence Tests is Equivalant to a Scoring Metric. March 2001. 11 pages.
ISBN 1 901615 61 8

24. Kaishev, V.K., Dimitrova, D.S., Haberman S., and Verrall R.J. Autumatic, Computer Aided Geometric Design of Free-Knot, Regression Splines. August 2004. 37 pages.
ISBN 1-901615-81-2

Faculty of Actuarial Science and Statistics

Actuarial Research Club

The support of the corporate members

CGNU Assurance
Computer Sciences Corporation
English Matthews Brockman
Government Actuary's Department
Swiss Reinsurance
Watson Wyatt Partners

is gratefully acknowledged.

ISBN 1-901615-81-2